

# 

Walter Becker, Ronald Böhmer, Birgitta Hauser, Klaus-Peter Luttkus, Carmen Rasche, Rainer Ross, Jörg Zeig

## IBM POWER Systems i Softwareentwicklung und -modernisierung

Handbuch für Strategien und Leitfaden für moderne Programmierung sowie Tipps & Tricks





## Web-Programmierung

RPG und Webanwendungen
HTML und System i
Allgemeine Angaben für den einfachen HTML-Export
HTML-Export unter Verwendung einer HTML-Schablone
Ferne Befehle
Ferne Befehle und RPG
Programmieren mit PHP
PHP on System i – Zend Core und Zend Platform
Download
Voraussetzungen i5/OS
PTFs
Ressourcen erstellen
Sicherungsdateien erstellen
Zend Core-Dateien entpacken
Zend-Objekte zum System i übertragen
Installation von Zend Core
Jobs im i5/OS
Funktionstest
Die Zend Core-Umgebung
Zend Platform installieren
PHP – Grundsyntax
Ein erster Schritt
Form und Stil
Datenbankzugriffe mit PHP
Die Aufgabe
PHP-Syntaxelemente
Variablen
Wertzuweisungen
Schleifen
Benötigte PHP-Funktionen
odbc_connect()
odbc_do() / odbc_exec()
odbc_fetch_row()
odbc_result()
odbc_close()



 _
<b>D</b>
Ť.
<b>D</b>

- 5.2.2.3.3 Das dokumentierte Script
- 5.2.2.3.4 Das undokumentierte Script
- 5.2.2.3.5 Die Ausgabe des Scripts
- 5.2.3 PHP und i5/OS-Befehle
- 5.2.3.1 runQry mit PHP? runQry mit PHP!
- 5.2.3.1.1 Vorbereitende Gedanken
- 5.2.3.1.2 Empfohlenes Vorwissen
- 5.2.3.1.3 Aufruf einer vorhandenen Abfrage
- 5.2.3.1.4 Datensätze auslesen (native)
- 5.2.3.1.5 HTML-Tabelle aus Datensätzen generieren
- 5.2.3.1.6 Restrukturierung und die ersten Klassen
- 5.2.3.1.7 Ein Webfrontend für den Query-Aufruf
- 5.2.3.1.8 Liste aller Queries in einer Bibliothek
- 5.2.3.1.9 Webfrontend mit Assistenten ausstatten.
- 5.2.3.1.10 Anzeigen von Bibliotheken, die Queries enthalten
- 5.2.3.1.11 Anzeigen der Abfragen in einer Bibliothek
- 5.2.3.1.12 Übernahme der Assistenten-Daten in den Befehlsprompt
- 5.2.4 DB2-Funktionen
- 5.2.4.1 Die Aufgabe
- 5.2.4.2 Benötigte DB2-Funktionen
- 5.2.4.2.1 Db2\_connect()
- 5.2.4.2.2 db2\_exec()
- 5.2.4.2.3 db2\_fetch\_row()
- 5.2.4.2.4 db2\_result()
- 5.2.4.2.5 db2\_close()
- 5.2.4.3 Das Script SVKLST00.PHP
- 5.2.5 Dialoge erstellen
- 5.2.5.1 Die Aufgabe
- 5.2.5.2 Exkurs: HTML-Formulare
- 5.2.5.2.1 Formularelemente
- 5.2.5.2.1.1 Einzeilige Eingabefelder
- 5.2.5.2.1.2 Radio Buttons
- 5.2.5.2.1.3 Submit- und Reset-Buttons
- 5.2.5.2.2 Das Suchformular: suchesvklst00.php

5

5.2.5.3	PHP: Parameterübergaben
5.2.5.3.1	Modifikation in "svklst00.php"
5.2.5.3.2	Script SVKLST00.PHP – V1.4
5.2.5.4	Ein anderer Lösungsansatz
5.2.5.5	Script SVKLST01.PHP – V1.0
5.2.5.6	Optimierungen 1
5.2.5.7	Exkurs Javascript
5.2.5.7.1	Einbettung
5.2.5.8	Funktionen
5.2.5.9	Optimierungen 2
5.2.5.10	Optimierungen 3
5.2.5.11	Optimierungen 4
5.2.5.12	Das Script SVKLST01.PHP – V1.1
5.2.5.13	Optimierungen 5: Ordnung schaffen
5.2.5.13.1	Das Script SVKLMASTER.PHP – V1.0
5.2.5.13.2	Das Script SVKLDISPLAY.PHP – V1.0
5.2.5.13.3	Das Script SVKLDBACCESS.PHP – V1.0
5.2.5.14	PHP-Funktionen einsetzen
5.2.5.14.1	PHP Funktionen
5.2.5.14.2	Parameter
5.2.5.14.3	Gültigkeit von Variablen
5.2.5.14.4	Rückgabewerte
5.2.5.14.5	Optionale Parameter
5.2.5.15	Datenbankzugriffsfunktionen erstellen
5.2.5.15.1	Die Funktion ConnectToDB2()
5.2.5.15.2	Die Funktion ExecuteDB2Sql()
5.2.5.15.3	Die Funktion DisplayDB2ResultSet()
5.2.5.15.4	Die Funktion CloseDB2Connection()
5.2.5.15.5	Das Script SVKLDBACCESS.PHP V1.1
5.2.5.15.6	Das Script db2functions.php
5.2.5.16	Das Formular optimieren
5.2.5.16.1	Exkurs: Was ist OOP?
5.2.5.16.2	Methoden
5.2.5.16.3	Eigenschaften
5.2.5.16.4	Klassen und Vererbung
5.2.5.16.5	Klassen erstellen
5.2.5.16.6	Aufruf der Klasse





- 5.2.5.16.7 Das Script: InputElementField.php
- 5.2.5.16.8 Das Script: InputElementRadiobutton.php
- 5.2.5.16.9 Das Script: InputElementButton.php
- 5.2.5.17 Stored Procedures einsetzen
- 5.2.5.17.1 Datenbankzugriff
- 5.2.5.17.2 Aufruf der Stored Procedures
- 5.2.5.17.3 Programmaufrufe mit Stored Procedures
- 5.2.5.17.3.1 Die Aufgabe
- 5.2.5.17.3.1.1 Grafischer Link
- 5.2.5.17.3.1.2 External Stored Procedure aufrufen
- 5.2.5.17.3.1.3 Die External Stored Procedure
- 5.2.5.18 Zend API für i5/OS
- 5.2.5.18.1 i5/OS Programmaufrufe
- 5.2.5.18.1.1 Die Aufgabe
- 5.2.5.18.1.2 Das Script svkldetailPC.php
- 5.2.5.18.1.3 Das Script svklPreDetail.php
- 5.2.5.18.1.3.1 Die Funktion i5\_connect
- 5.2.5.18.1.3.2 Die Funktion i5\_program\_prepare
- 5.2.5.18.1.3.3 Die Funktion i5\_program\_call
- 5.2.5.18.1.3.4 Die Funktion i5\_close
- 5.2.5.18.1.3.5 Mit PCML arbeiten
- 5.2.5.18.1.4 Das Script svklDisplayPC.php
- 5.2.5.18.1.5 Das Script svklPostDetail.php
- 5.2.5.19 PHP und Opensource
- 5.2.5.19.1 Menüs mit CoolJs generieren
- 5.2.5.19.2 Die Installation von MySQL
- 5.2.5.19.2.1 Download MySQL
- 5.2.5.19.2.2 Benutzerprofil erstellen
- 5.2.5.19.1.3 Installationsdateien entpacken
- 5.2.5.19.2.4 Entpackte Dateien auf das System i kopieren
- 5.2.5.19.2.5 PASE-Terminal öffnen
- 5.2.5.19.2.6 Tar-File entpacken
- 5.2.5.19.2.7 Logischen Link erstellen
- 5.2.5.19.2.8 Installationsscript ausführen
- 5.2.5.19.2.9 MySQL Probestarten
- 5.2.5.19.3 MySQL produktiv nutzen
- 5.2.5.19.3.1 Jobwarteschlange MySQL



- 5.2.5.19.3.2 Ausgabewarteschlange MySQL
- 5.2.5.19.2.3 Jobbeschreibung MySQL
- 5.2.5.19.3.4 Jobklasse MySQL
- 5.2.5.19.3.5 Subsystem MySQL
- 5.2.5.19.3.6 Start des MySQL-Servers
- 5.2.5.19.3.7 Beendigung des MySQL-Servers
- 5.2.5.19.3.8 Administration
- 5.2.5.19.4 Mediawiki installieren
- 5.2.5.19.5 SugarCRM installieren
- 5.2.5.19.5.1 Voraussetzungen:
- 5.2.5.19.5.2 Installation
- 5.2.5.19.5.3 SugarCRM aufrufen
- 5.2.5.19.5.4 Anfangskonfiguration
- 5.3 Das Kapitel wird überarbeitet.

#### 5.4 Das Kapitel wird überarbeitet.





5.5	Web-Anwe	ndu	ngen	auf	dem	HTT	P-Serv	ver
1		17	c					

- 5.5.1 HTTP-Server-Konfiguration
- 5.5.2 CGI-Anwendungen
- 5.5.3 Integrierte Web Services
- 5.5.3.1 Vergabe eines Server-Namens für Web Services
- 5.5.3.2 Angabe der Benutzer-ID
- 5.5.3.3 Deployen eines RPG-Programms als Web Service
- 5.5.3.4 Details zum IBM i-Programm
- 5.5.3.5 Namensvergabe für den Web Service
- 5.5.3.6 Auswahl der Exportprozeduren
- 5.5.3.7 Angabe der Benutzer-ID
- 5.5.3.8 Angabe der Bibliotheksliste
- 5.5.3.9 Zusammenfassung
- 5.5.4 Verwaltung der definierten Web Services
- 5.5.5 Node JS und System i
- 5.5.6 REST-Webservices
- 5.5.6.1 Verwalten von REST-Webservices
- 5.5.6.2 REST-Webservice definieren
- 5.5.6.3 REST-Webservice testen
- 5.5.6.4 REST-Serviceprogramm Kundenstamm
- 5.5.6.4.1 Installation des Webservices
- 5.5.6.4.2 Webservice Kunden löschen
- 5.5.6.4.3 Webservice Kunden-Update
- 5.5.6.4.4 Webservice Kunde erstellen
- 5.5.6.4.5 Webservice Kundenabfrage einzeln
- 5.5.6.4.6 Webservice Kundenabfrage gesamt
- 5.5.7 SOAP Webservices
- 5.5.7.1 Details zum IBM-i-Programm
- 5.5.7.2 SOAP Webservice erstellen
- 5.5.7.2.1 Namensvergabe für den Webservice
- 5.5.7.2.2 Auswahl der Exportprozeduren
- 5.5.7.2.3 Angabe der Benutzer-ID
- 5.5.7.2.4 Angabe der Bibliotheksliste
- 5.5.7.2.5 Zusammenfassung
- 5.5.7.3 Verwaltung der definierten Webservices







Kapitel



#### 5.1 Seite 1

## 5.1 RPG und Webanwendungen

Moderne Anwendungen werden heute häufig in einem Atemzug mit Webanwendungen genannt. Die Allgegenwärtigkeit des Internets und die Verbreitung der damit verbundenen Technologien ist enorm. Folglich liegt es auf der Hand, dass eine Reihe von Anwendungen auf eben diesen Technologien basieren.

Das Betriebssystem IBM i unterstützt seit Jahren die Verwendung von Internetfunktionen. Sei es als integrierter Webserver, der in Form des Apache Webservers auf dem System die Grundlage für eine Reihe von Standardfunktionen bietet, oder seien es die verschiedenen Werkzeuge, mit denen IBM i-Betreiber die Integration bestehender DB2/UDB-Daten in Webanwendungen realisieren können.

Leider haben die unterschiedlichen Werkzeuge und Anwendungen nur sehr wenig Akzeptanz erfahren. Das mag hier und da auch verständlich erscheinen, ist deren Umsetzung doch teilweise sehr komplex.

Geht man von einem "typischen" IBM i-Betreiber aus, so ist festzustellen, dass hier meistens ein solides Wissen um die Datenbank und im Umgang mit RPG & Co vorhanden ist. Dieses Wissen, kombiniert mit den Möglichkeiten der Webanwendungsentwicklung kann eine sehr nützliche Konstellation ergeben.

Stellen wir uns einmal vor, dass ein Unternehmen seine DB2/UDB-Daten zusammen mit einer modernen Webanwendung, beispielsweise einem Webshop, kombinieren will.

Sie werden vermutlich einige Technologien nennen können, mit denen eine solche Umsetzung durchgeführt werden kann. Wir wollen uns jedoch keiner Zusatztools bedienen, sondern die Umsetzung mit RPG-Mitteln bewerkstelligen.





Grobablauf Webanwendung

Bestätigen Zahlung / Ablehnung Zahlung

In dieser Beispielanwendung wird ein klassischer, auf HTML-Basis betriebener Webshop ausgeführt. Einige der Daten liegen in der Datenbank des Systems IBM i, etwa der Artikelstamm, der Lagerbestand und die Reservierungsdaten. Diese Tabellen müssen von der Webanwendung eingelesen und zum Teil auch fortgeschrieben werden.

Nachdem die Artikelauswahl abgeschlossen ist, soll die Bestellung durch eine Zahlungsprüfung abgeschlossen werden. Dazu werden Kreditkartendaten geprüft und/oder die Dienste eines Online-Zahlungssystems wie z. B. Paypal in Anspruch genommen. Je nach Ausgang der Prüfung wird anschließend eine Bestätigung oder Ablehnung an die aufrufende Anwendung zurückgegeben. Erst wenn diese Transaktion abgeschlossen ist und der Kunde seine Zahlungsinformationen eingegeben hat, diese geprüft und als korrekt zurückgegeben wurden, kann die Weiterverarbeitung der Bestellung erfolgen. Ergo: eine typische Webanwendung.



**RPG und Webanwendungen** 

Betrachten wir einige Beispiele, mit denen eine Einbindung der IBM i-Daten in eine Webanwendung umgesetzt werden könnten:

- Webanwendung verwendet Schnittstellen für den Zugriff auf die DB2. Es bedarf einer sorgfältigen Sicherheitsanalyse und Sicherheitsumsetzung, damit auch wirklich nur auf die gewünschten Informationen zugegriffen werden kann.
- WebFacing

WebFacing ist als Einstieg in die Welt der IBM i und in Webanwendungen insofern ideal, da man basierend auf DDS-Beschreibungen zusätzlich zu den 5250-Bildschirmen auch Webseiten generieren und einsetzen kann. Die Nachteile liegen in den teils statischen Anwendungen und in Einschränkungen, die sicherheitsrelevant sein können (z. B. in der Notwendigkeit, dass für die Ausführung von WebFacing auf Benutzerebene die Einstellung "Möglichkeiten einschränken" auf \*NO stehen muss). Dennoch ist WebFacing als Schnelleinstieg eine Überlegung wert.

• HATS

Ähnlich wie mit WebFacing bietet IBM mit HATS ein Produkt an, mit dem IBM i-Anwendern der Zugriff auf die DB2-Daten mit Webanwendungen ermöglicht wird. Im Gegensatz zu WebFacing ist hier mehr Wissen von den Anwendungsentwicklern gefragt. Das gilt zumindest dann, wenn HATS nicht nur in seiner Grundform, d. h. "von der Stange" eingesetzt werden soll.

- Diverse Tools von Drittanbietern
- Nutzung von Java
- Nutzung von PHP
- Nutzung von RPG

Diese Auflistung ist selbstverständlich nicht vollständig und stellt lediglich einige der Möglichkeiten vor, mit denen DB2/UDB-Daten für Webanwendungen verfügbar gemacht werden können.

In diesem Beitrag möchte ich Ihnen eine Variante der Integration von Webanwendungen in IBM i-Anwendungen vorstellen, die auf RPG basiert. Mit ein wenig Geschick und den entsprechenden Kenntnissen kann man in der Tat eine Webanwendung mit RPG steuern und die Daten auch für IBM i-Zwecke nutzen. Natürlich wird der eine oder andere Leser nun denken, dass eine solche Umsetzung doch ein wenig am Ziel vorbeiführt. In der Tat sind die zuvor genannten Alternativen allesamt in der Lage, Webanbindungen umzusetzen. Doch nicht alle System i-Betreiber verfügen über Mitarbeiter, die sich mit Sonderthemen oder Entwicklungssprachen wie PHP oder Java auskennen.

**RPG und Webanwendungen** 

5.1 Seite 4 RPG hat sich in den letzten Jahren stark verändert und von einer doch recht eingeschränkten Programmiersprache zu einem mächtigen Entwicklungswerkzeug gewandelt. Die Nutzung aller mit RPG IV und ILE zur Verfügung stehenden Möglichkeiten gestattet es IBM i-Betreibern nicht nur, bestehende Datenbankkomponenten unverändert zu nutzen, sondern ermöglicht auch die Integration modernster Datenbankbereiche sowie die Entwicklung wirklich moderner Anwendungen für das System i-Umfeld.

Um die Webanbindung mit RPG umsetzen zu können, bedarf es einer gewissen Grundkenntnis im ILE-Umfeld. ILE als Abkürzung für "integrated language environment" bildet die Grundlage für das Öffnen der AS/400-Anwendungen hin zu modernen IBM i-Applikationen. Zudem muss beachtet werden, dass die Datenhaltung auf dem System i von der Datenhaltung auf anderen Systemen abweicht. Hierbei ist nicht nur die Besonderheit der relationalen Datenbank DB2/UDB zu nennen, sondern auch die Tatsache, dass neben der Verwendung des Zeichensatzes Internationalität berücksichtigt werden muss. Letzteres zumindest dann, wenn ein Webshop multilingual zum Einsatz kommen soll. Es ist anderenfalls ärgerlich, wenn Sonderzeichen aus anderen Alphabeten fehlerhaft dargestellt werden. Erschwerend kommt hinzu, dass viele Webanwendungen zur Absicherung der eingegebenen Daten bestimmte Verschlüsselungsverfahren anwenden, Verschlüsselungsverfahren, die leider sehr empfindlich sind, was die korrekte Zeichensatzverwendung betrifft. Doch dazu später mehr.

Eine gewisse Grundkenntnis in HTML ist ebenfalls von Vorteil, wird doch ein Großteil der Kommunikation im Webbereich auf Basis von HTML durchgeführt. Das Gleiche gilt für XML. Aber keine Angst – auch wenn diese Technologien natürlich in Verbindung mit einer Webanbindung zu nennen sind, an dieser Stelle sind wirklich nur Grundkenntnisse vonnöten, die zum Teil im Folgenden auch aufgezeigt und erläutert werden.

Verfeinern wir unser Beispiel ein wenig und schauen uns an, wann und wo exemplarisch RPG-Komponenten in den Ablauf eingebunden werden können. Diese RPG-Komponenten dienen dem Einlesen der IBM i-Daten und bei Bedarf auch dem Fortschreiben derselben:



**RPG und Webanwendungen** 



Seite 5



RPG und HTML

In der Beispielanwendung ruft ein Standard-RPG-Auftragserfassungsprogramm eine Webshopanwendung auf. Während das RPG-Auftragserfassungsprogramm klassisch in IBM i-eigenen Technologien wie z. B. CL und RPG entwickelt wurde, stellt der Webshop eine quasi "IBM i-fremde" Welt dar. Bei diesem Webshop kann es sich durchaus um einen Standardwebshop eines Drittanbieters handeln, der beispielsweise in Java, XML, HTML oder sonstigen gängigen Technologien erstellt wurde.

Hier haben wir bereits eine erste Schnittstelle zwischen IBM i-Anwendung und WebShop. Nehmen wir einmal an, dass die RPG-Anwendung neben einer Firmennummer auch die Auftragsnummer, den Kundennamen und weitere Details zu einer Bestellung an den Webshop übermittelt.

Die erste Frage ist nun, wie kann ein RPG-Programm Informationen an eine Webanwendung senden? Die Antwort lautet ganz einfach: mit Standard-RPG-Funktionen.





Ablauf Webshop-Anbindung

Bearbeiten Zahlungsinformationen

In dem Webshop wird basierend auf den mitgegebenen Eingabeparametern die Weiterverarbeitung durchgeführt. Der Kunde kann Artikel auswählen, deren Details anschauen und Verfügbarkeits- und Preisabfragen durchführen. Diese lassen sich durchaus mit RPG-Funktionen abbilden.



Java / HTML / Webanwendung

Bei dem Tool HTTPAPI handelt es sich um ein Open Source Tool – Quellen sind also vorhanden und unterliegen einer permanenten Weiterentwicklung.

#### IBM POWER Systems i Softwareentwicklung und -modernisierung

Nachdem der Kunde seine Artikelauswahl abgeschlossen hat, wird die Zahlung durchgeführt. Als Beispiel: der Kunde wählt als Zahlungsart die Kreditkarte und gibt seine Kreditkarteninformationen ein. Natürlich muss das gesichert erfolgen. In der Webanwendung werden diese Teilschritte abgewickelt. Nach der erfolgten Zahlung soll der Webshop nun die Verarbeitungsinformationen zurück in die Datenbank des IBM i schreiben. Dazu bedienen wir uns wieder Standard-RPG-Funktionen.

Natürlich ist dieser Beispielablauf hier stark vereinfacht. Er zeigt jedoch vom Grundsatz her die gewöhnliche Abwicklung eines Einkaufs im Internet.

Nachfolgend möchte ich Ihnen nun zeigen, wie RPG dazu verwendet werden kann, um

- 1. Webdaten auszulesen
- 2. Webdaten einzulesen

Ein grundsätzliches Thema bei der Einbindung von Webseiten in System i-Anwendungen ist die Thematik des Auslösers. Gemeint ist hier die Art des Aufrufs bzw. des Anwendungsflusses. Hier sind die folgenden Fälle zu unterscheiden:

- Aufruf einer Webanwendung von einer IBM i-Anwendung aus
- · Aufruf einer IBM i-Anwendung von einer Webanwendung aus

In diesem ersten Beispiel werden wir uns auf die Initiierung einer Webintegration aus einer IBM i-Anwendung heraus beschränken. Der Aufruf einer IBM i-Anwendung aus einer Webanwendung heraus folgt in einem weiteren Beispiel.

Die Grundlage für die Entwicklung sind Standard-ILE RPG-Komponenten. Diese können wahlweise in Fix- oder Freeformat genutzt werden.

Man muss nicht das Rad neu erfinden, um eine RPG-Anwendung in die Lage zu versetzen, Daten an eine Webseite senden und Informationen aus einer Webseite abrufen zu können. Ich verwende für solche Anforderungen eine Freeware, die in IBM i-Kreisen weit verbreitet ist. Scott Klement, AS/400-"Enthusiasten" durch eine Vielzahl von Vorträgen und Veröffentlichungen bekannt, stellt auf seiner Webseite eine Fülle an wertvollen Informationen rund um das System und dessen Anwendungsentwicklung zur Verfügung.

Die hier verwendeten Module und Programme finden Sie zum kostenlosen Download auf der folgenden Webseite:

http://www.scottklement.com/httpapi/

RPG und Webanwendungen





Das verdeutlicht auch stark vereinfacht die folgende Abbildung:



Informationsfluss Webanwendung Browser



Schauen wir uns die Abbildung oben einmal genauer an:

Im Normalfall startet ein Anwender einen Browser und gibt eine URL zum Aufruf einer bestimmten Webseite ein. In diesem Kommunikationszweig bietet der Browser die Schnittstelle zwischen Anwender und Webanwendung. Unabhängig davon, welche Details die Webanwendung aufweist, können dort in der Regel Eingaben vorgenommen werden. Diese werden dann an die Webanwendung gesendet, dort verarbeitet und führen zu einem Ergebnis, das mit einer Datenrückgabe an den Browser und der dortigen Darstellung abgeschlossen wird.

Für die Einbindung einer Webanwendung kann der Kommunikationsfluss im Groben ähnlich abgebildet werden. Die Komponente "Browser" wird ersetzt. Die dort eingebundenen Funktionen (Parameter- oder Dateneingabe sowie das Zurückübertragen der Ergebnisinformationen) übernimmt in diesem Fall das HTTPAPI. Das soll die folgende Skizze verdeutlichen.



Seite 9



Informationsfluss Webanwendung RPG



An Stelle des Browsers sehen Sie hier das HTTPAPI. Es übernimmt die Aufgabe der Übertragung der Eingabeinformationen an eine Webseite und ist ebenso in der Lage, die zurückgegebenen Werte und Informationen der IBM i-Anwendung – hier dem RPG-Programm – wieder zur Verfügung zu stellen.

Für die Kommunikation zwischen RPG und dem Internet wird das HTTP-Protokoll verwendet und in eigens dafür entwickelten Service-Programmen verarbeitet. Damit eine Webanwendung die IBM i-Daten empfangen bzw. Daten in die DB2/UDB stellen kann, müssen diese Daten entsprechend vorliegen. Dazu kann beispielsweise HTML und/oder XML verwendet werden.

Das Paket zum Herunterladen der RPG-Web-Fähigkeit umfasst neben den ausführbaren Objekten auch den Quellcode und einige wirklich sehr sinnvolle und leicht verständliche Beispiele.

Das HTTPAPI-Paket umfasst eine wahre Fülle an Funktionen. Beispielsweise ist es damit auch möglich, sichere Verbindungen mittels SSL herstellen und ausführen zu können. Die Integration der SSL-Standards ist in den Komponenten bereits enthalten.

Ein bedeutender Teilschritt bei der Implementierung von Webseiten in eine RPG-Anwendung liegt in der korrekten Übermittlung der Daten. Die Besonderheit der Datenhaltung in der relationalen Datenbank DB2/UDB liegt u.a. im verwendeten EBCDIC-Zeichensatz begründet. Webanwendungen sind in der Regel nicht in der Lage, diese Daten in jedem Fall 1:1 zu verstehen. Erschwerend kommen noch die unterschiedlichen internationalen Einstellvarianten hinzu, welche die korrekte Umsetzung von Sonderzeichen erschweren.

Im Zuge der Internationalisierung hat IBM diesen natürlich auch Rechnung getragen und bietet in Verbindung mit RPG die Option, Daten im allgemeingültigen UTF-Format verarbeiten zu können. Das UTF-Format bietet sich für die meisten Webanwendungen an. Details zu UTF finden Sie in anderen Fachpublikationen im Zusammenhang mit der Administration des System i.

An dieser Stelle sei darauf hingewiesen, dass bei der Konzeption einer Webanbindung mit RPG ggf. Zeichenumsetzungen oder eine entsprechende Verwendung eines Zeichensatzes erforderlich sein können.

Lassen Sie uns mit einem einfachen Beispiel beginnen, dem ich noch einen kleinen Exkurs in die Welt von HTML voranstellen möchte.

Die Kommunikation zwischen dem System i und der Welt des Internets sollte sich nach Möglichkeit auf Basis der geläufigen Webstandards bewegen. HTML als Grundlage für einen Datenaustausch ist sicher in vielen Fällen ideal. Auch XML kann hier durchaus eingesetzt werden.

Seite 11



RPG und Webanwendungen

5.1	
Seite	12

Schauen Sie sich das folgende Beispiel einer wirklich sehr einfachen Webseite an:



Webseite

Diese Abbildung zeigt das Ergebnis des folgenden HTML-Textes:

```
<html>
<head>
<title>Beispiel einer einfachen Webseite</title>
</head>
<body>
<h1>Das ist eine ganz einfache Webseite in HTML.</h1>
<h2>Erstellt von H.Jörg Zeig</h2>
Ich habe diese zur Veranschaulichung erstellt. Es gibt
keine Funktionen -
lediglich eine einfache Anzeige in einem Browser.
 ....und noch ein Bild eingebunden.... 
<img src="http://www.midrangemagazin.de/pms/techknow/pics/
head2.gif" />
```



RPG und Webanwendungen



Dieses sehr einfache HTML-Dokument beinhaltet verschiedene Stufen von HTML.

Den Bodybereich nebst Titel und unterschiedlichen Dokumentenebenen, die sich u. a. durch verschiedene Schriftgrößen differenzieren.

Wie Sie sehen, findet sich für jede HTML-Startanweisung (in eckige Klammern gefasst), auch eine abschließende HTML-Anweisung (gekennzeichnet durch die Syntax </xx>).

Neben den unterschiedlichen Textelementen ist hier auch eine externe Bildquelle eingebunden, die auf ein Bild verweist, das in einem Unterverzeichnis von www.midrangemagazin.de zu finden ist.

So einfach dieses statische HTML-Dokument auch ist, es beinhaltet von der Grundlage her einige wesentliche HTML-Bausteine.





Web-Programmierung RPG und Webanwendungen

5.1 Seite 14





#### 5.1.1 Seite 1

## 5.1.1 HTML und System i

Heterogene Anwendungen bedienen sich meist allgemeiner Standards. Es verwundert daher nicht, dass sich auch viele "moderne" Anwendungen, wie zum Beispiel Webanwendungen, dieser Standards bedienen und im Umfeld des System i Anwendung finden. HTML ist in diesem Zusammenhang bekanntlich ein wesentlicher, ja allgegenwärtiger Baustein und wird natürlich auf verschiedene Weisen auch mit dem System i unterstützt, beispielsweise beim Einsatz von WebServern. So vielfältig die Möglichkeiten des Einsatzes von HTML mit dem System IBM i sind, so komplex stellen sie sich allerdings für den IBM i-Betreiber dar. Doch das Buch mit sieben Siegeln kann manchmal so einfach zu öffnen sein …

Ich verzichte an dieser Stelle bewusst auf die Grundlagen von HTML. Diese finden Sie zur Genüge im Internet oder in speziellen Fachbüchern dokumentiert. Wir wollen hier vielmehr gezielt auf die Bereitstellung von DB2/UDB-Daten im HTML-Format eingehen. Eine Anforderung, die zum Beispiel bei der Kommunikation mit Kunden oder Lieferanten oder bei der Anbindung eines Webshops gegeben ist.

Wir können unsere System i-Daten in unterschiedlichen Formen für eine Webanwendung bereitstellen. Neben verschiedenen ausgefeilten Tools, die mittels Vorgaben oder individueller Programmierung den Inhalt von DB2-Dateien oder -Tabellen in HTML umsetzen, können wir uns auch vergleichsweise einfacher Werkzeuge bedienen, die eine Grundform von HTML realisieren.

In diesem Kapitel möchte ich Sie mit einer wirklich sehr einfachen Möglichkeit der Darstellung von System i-Daten in einer Webanwendung bekannt machen, der Exportfunktion der Client Version des Lizenzprogramms System i Access für Windows (beziehungsweise der Exportfunktion einer der Vorgängerversionen, wie zum Beispiel iSeries Access für Windows), die bei den meisten Installationen eh zur Verfügung steht.

Sicher kennen Sie diese Exportfunktion, mittels derer wir in der Lage sind, System i-Daten aus einer DB2-Datenbank in ein PC-Format wie zum Beispiel Excel zu exportieren. HTML und System i

#### 5.1.1 Seite 2

#### 5.1.1.1 Allgemeine Angaben für den einfachen HTML-Export

Die Darstellung von Daten in HTML kann auf unterschiedliche Weisen erfolgen. Im einfachsten Fall exportieren wir bestehende Daten weitgehend "unformatiert" nach HTML. Dabei beeinflussen wir lediglich einige wenige Darstellungsmerkmale der HTML-Ausgabe. Das Ergebnis ist eine mehr oder weniger strukturierte, anzeigenoptimierte Ausgabe der IBM i-Daten in HTML-Form. Welche Einstellungsoptionen hierbei möglich sind, zeige ich Ihnen im nachfolgenden Beispiel, in dem wir den Inhalt einer IBM i-Datei (Kundenstamm) zwecks Darstellung in eine Webseite exportieren. Dabei verzichten wir zunächst auf eine optimierte Anzeigenaufbereitung und konzentrieren uns ganz auf den Export der IBM i-Daten ins HTML-Format.

Für den Export sind zunächst einige Pflichtangaben zu machen, die wir auf der Einstiegsseite der "Datenübertragung von System i" finden und die Sie vermutlich schon einmal vornehmen mussten, als Sie einen Export von System i-Daten in Excel- oder Textdateien durchgeführt haben. Genau denselben Assistenten nutzen wir in unserem ersten HTML-Export-Beispiel.

Sollten Sie diese Funktion noch nicht kennen – sie lässt sich über den Programmblock "IBM System i Access für Windows" öffnen. Wählen Sie darin den Eintrag "Datenübertragung von System i".

Die f	olgend	le Abbil	dung ze	eigt die	Datenü	bertragu	ngsanze	eige:

🖳 Datenübertra	gung von System i		8 <u>-   X</u>
<u>D</u> atei <u>A</u> nsicht <u>H</u> i	lfe		
🏠 🖻 🖬 😭			
	System i System i-Name: Dateiname:	aszebis zeigrpg/kunden 1	Durghsuchen
	PC <u>A</u> usgabeeinheit: Da <u>t</u> einame:	HyperText Markup Language (HTML)	Details
		Daten von System i übertragen	Formatoptionen

Datenübertragung System i Access



HTML und System i

In dieser Anzeige muss der Hostname des System i oder alternativ dessen TCP/ IP-Adresse angegeben werden.

Im Bereich "Dateiname" muss der qualifizierte Name der zu exportierenden System i-Datei (Bibliothek/Dateiname) angegeben werden. In unserem Beispiel verwenden wir die Datei "Kunden1". Sie beinhaltet die Kundenstammdaten.

Im unteren Bereich der Anzeige wählen wir als Ausgabeeinheit "Hyper Text Markup Language (HTML)" und geben Pfad und Namen der neu zu erstellenden HTML-Datei an.

Idealerweise geben wir nach dem Dateinamen auch die Dateiart an – also "html". Damit erleichtern wir nachfolgend eingesetzten Anwendungen das Handling mit unserer HTML-Exportdatei.

Die Pflichtangaben für den Export nach HTML können durch verschiedene Optionen ergänzt werden, individuelle Einstellungen, die zum Beispiel das Aussehen der HTML-Ausgabe beeinflussen oder deren Inhalt näher definieren. Dazu klicken wir auf die Schaltfläche "Details".

HTML-Details			<u>a</u> : ×						
Zeile Datei	Zelle Dokument	Tabelle	Schablone Beschriftung						
Automatisch Maximale D	ne Größe verwende ateigröße: 128	n KB							
Bestätigung	vor Überschreiben	ı							
Umsetzung ur	nd Zeichensatz								
System i-Daten	System i-Daten umsetzen in C ANSI C ASCII C UTF-8								
Zeichensatz:	utf-8								
ОК	Abbrechen	Übernehr	nen Hilfe						

Detailauswahlen für den HTML Export

Ergänzung 29/2014

Seite 3



HTML und System i

5.1.1 Seite 4 Die in der HTML-Detailanzeige verfügbaren Optionen sind in unterschiedliche Auswahlbereiche separiert.

#### Der Bereich "Datei"

Im Bereich "Datei" lassen sich Vorgaben für die Erstellung der Zieldatei hinterlegen. Die Angabe einer maximalen Größe für eine neue HTML-Datei ist zum Beispiel erforderlich, weil einige Geräte und Tools, wie zum Beispiel Handhelds oder auch bestimmte Webanwendungen, eine Größenlimitierung für zu verarbeitende HTML-Dateien verlangen.

Die wichtigste Angabe betrifft die gegebenenfalls erforderliche Umsetzung des Zeichensatzes. Die Besonderheiten der Zeichen- bzw. Datenspeicherung auf unserem System i bedingen eine Umsetzung in ein nicht-IBM-i-lesbares Format. Das betrifft insbesondere die Verwendung bzw. Darstellung von Sonderzeichen und Umlauten. Letztlich kommt es darauf an, welche Form der HTML-Ausgabe genutzt werden soll oder wie und wo die HTML-Verarbeitung stattfindet. In der Regel kann man mit einer Umsetzung in UTF-8 arbeiten, da dieser Zeichensatz von Webanwendungen und Browsern unterstützt wird.

#### Der Bereich "Dokument"

Der Bereich "Dokument" erlaubt die Angabe allgemeiner Informationen, die in die HTML-Datei eingebunden werden sollen. Unter anderem lassen sich hier der Titel sowie Datum und Uhrzeit angeben.

HTML-Details					<u>⊜</u> ?×
Zeile Datei	) Dokumer	Zelle t	Í Tabelle	Sch   B	nablone   eschriftung   
<u>T</u> itel:	Kunden				
- Datum/Ze	it				
🔽 Įnklusi	ve Datum und	Zeit			
Position fü	ir Datum/Zeit:	• <u>O</u> be	en links	O O <u>b</u> er	n rechts
		0 <u>U</u> nt	en links	O U <u>n</u> te	n rechts
O	( Abb	rechen	Ü <u>b</u> err	nehmen	Hilfe





		-				
W	/eb-	Pro	ara	mmi	erun	a
						-

Kapitel

5.1.1

Seite 5

HTML und System i

Wie Sie in der nächsten Abbildung sehen, ist der Titel im Browserbereich angegeben. Das Datum und die Uhrzeit wurden gemäß unserer Vorgaben im oberen linken Bereich der HTML-Ausgabe platziert.

Kunden - Wind	ows Internet	Explorer provided							8_0
) 🕑 - 🥑 🕻	C:\Temp\kunden	1.html 🔎 🗸 🐓	🗙 🌽 Kunden	1	×				🕯 😒
Donnerstag, 13 2:42:24	. März 2014								
KUMANDT	KUKDNR	KUNAME	KUSTRASSE	KUPLZ	KUORT	KUSTATUS	KUUUSER		
01	2	Willi Winzig	Hauptstrasse 1	57071	Siegen	A			
01	10	Helmut Kohl	Dorfstrasse	60000	Ockersdorf	A	]		
)1	15	Heidi Khum	Bergstrasse	54880	Bergisch Gladbach	A	]		
)1	20	Otto Kabel	Mainstrasse 7	50001	Köln	A			
)2	2	Willi Winzig	Poststrasse 18	57482	Olpe	А			
)2	90	Siegfried Schmidt	Dorfweg 7	61145	Walddorf	A			

Dokumentangaben – Auswirkung





HTML und System i

5.1.1 Seite 6

#### Der Bereich "Tabelle"

Die HTML-Ausgabe einer Datei erfolgt meist in Form einer Liste, die aus mehreren Zeilen besteht. Mit den Angaben im Abschnitt "Tabelle" können wir die Positionierung der Tabelle im Anzeigenfenster bestimmen und zum Beispiel festlegen, ob Spaltenüberschriften zur besseren Identifizierung des Inhalts des HTML-Dokuments verwendet werden sollen. Werden keine weiteren Angaben gemacht, so wird basierend auf der IBM i-Datei für jedes Feld (bzw. jede Spalte) eine Spalte in der HTML-Ausgabe erzeugt. Ähnlich verhält es sich mit der Einbindung von Zeilen oder auch der Darstellung der Umrandung der Tabellenanzeige.

HTML-Details					<u>8</u> ?×
Zeile Datei	Dokumen	Zelle t	j Tabelle	Scha Be:	blone schriftung
- Ausrichtun C Lini	g ks	• Mitt	e	0 [	Rechts
Attribute	ve Spaltenübe	rschriften			
C Anzahl	<u>Z</u> eilen: <u>S</u> palten:	2 2	A 1	Optic	men
E Rahme	en <u>b</u> reite (Pixel) abstand (Pixel)	: 1			
☐ Zellenr ☐ <u>T</u> abelle	andbreite (Pi <u>x</u> e enbreite:	el): 1 100		% des F	enstei 🔻
OK	Abb	rechen	Übem	ehmen	Hilfe

Tabellenangaben

HTML und System i



Seite 7

#### Der Bereich "Beschriftung"

Nun fehlt uns noch die "Überschrift" zu unserem HTML-Export. Diese können wir im Bereich "Beschriftung" festlegen.

TML-Details			<u> </u>
Zeile	Zell	e	Schablone
Datei	Dokument	Tabelle	Beschriftung
– Beschriftung			
Text: Au	sgabe Datei KUN	DE in Webseite	
	Inklusive Tabellei	nnummer	
Attribute			
<u>A</u> usrichtung:	Standardwert	-	
Textgröße:	Nein	C <u>N</u> ormal	C <u>G</u> roß
Darstellung:	<mark>⊡</mark> <u>S</u> tandard	🗖 <u>F</u> ett	🗖 Kursi <u>v</u>
	🗖 Unterstrichen	Feste <u>B</u> re	ite
		1	
OK	Abbreche	en Ü <u>b</u> erneh	imen Hilfe

#### Beschriftung

Über die Option "Inklusive Tabellennummer" wird die Nummer der in HTML eingebundenen Tabelle gegebenenfalls automatisch hochgezählt. Das macht dann Sinn, wenn man mit mehreren Tabellen in einer HTML-Ausgabe arbeitet.

Mit Hilfe der Attribute können wir noch auf die Textgröße oder die Form der Textdarstellung für den Überschriftsbereich Einfluss nehmen. Hier können wir unter anderem Fettschrift oder Darstellung in kursiver Form auswählen.



HTML und System i

5.1.1 Seite 8

#### Der Bereich "Zeile"

Mit der Übertragung der System i-Datei wird für jeden Datensatz der Datei eine Zeile in der HTML-Ausgabe generiert. Der Auswahlbereich "Zeile" erlaubt uns die Definition der Darstellung sowie der Positionierung der Zeilen in unserer HTML-Ausgabe:

HTML-Details	<u> </u>
Datei Dokument Zeile Zelle	Tabelle Beschriftung
Attribute für Header Horizontale Textausrichtung:	Mitte
Vertikale Te <u>x</u> tausrichtung:	Standardwert 💌
Textdarstell. 🔽 Standard	<u>F</u> ett <u>F</u> ett <u>Feste Breite</u>
Attribute für allgemeine Zeilen -	
Horizontale Textausrichtung:	Standardwert 💌
Vertikale Textausrichtung:	Standardwert 💌
Textdarstell.: 🔽 Standard	Fett     Feste Breite
OK Abbreche	n <u>Üb</u> ernehmen Hilfe

#### Zeile

Wie Sie anhand der Abbildung oben sehen können, unterscheidet IBM an dieser Stelle zwischen Angaben zu Headerdetails und zu allgemeinen Zeilen.



HTML und System i



Seite 9

#### Der Bereich "Zelle"

Ähnlich wie wir Zeilenattribute für die HTML-Ausgabe definieren können, lassen sich auch Angaben für die Zellen hinterlegen. Diese Angaben beschränken sich auf die Ausrichtung der Zellen (Links, Rechts, Mitte), und zwar differenziert nach alphanumerischen Werten und Zahlenwerten.

TML-Details			<u>⊜</u> ?×
Datei Zeile	Dokument Zelle	Tabelle	Beschriftung Schablone
Datenausric <u>T</u> extdaten: <u>N</u> umerische	htung Zeilenstar Daten: Zeilenstar	ndardwert	
Attribute Textgröße: IZ Text um	• Klein C	Normal (	<u>G</u> roß
OK	Abbrecher	n Ü <u>b</u> emeh	men Hilfe

#### Zelle

Die Differenzierung in alphanumerisch und numerisch macht durchaus Sinn denn im Allgemeinen verwenden wir auch eine Linksausrichtung für Textfelder, wohingegen Zahlenfelder in der Regel rechtsbündig ausgerichtet werden.

Des Weiteren kann die Textgröße über eine Grobauswahl festgelegt werden. Sollte ein Text eine gewisse Länge überschreiten, so kann mit der Option "Text umbrechen" ein Textumbruch erzwungen werden.

Damit sind die wesentlichen Angaben für einen Export und eine mehr oder weniger gute Darstellung im HTML-Basisformat abgeschlossen. Schauen wir uns das Ergebnis einmal an.

Nachdem Sie die gewünschten Änderungen an den HTML-Details vorgenommen haben, klicken Sie auf "Übernehmen" und dann auf "OK". Damit gelangen wir wieder in die Ausgangsanzeige zum Export von System i-Daten.



Ergänzung 29/2014



HTML und System i

#### 5.1.1 Seite 10

🖳 Datenübertra	gung von System i	i	8 <u>- </u> ×
Datei Ansicht Hil	fe		
11 🖻 🖬 😭			
	− System i S <u>y</u> stem i-Name: Dateiname:	172.16.100.1 zeigrpg/kunden	Durghsuchen
			Datenoptionen
<sup>0</sup> 0	PC		
00	<u>A</u> usgabeeinheit:	HyperText Markup Language (HTML)	Details
	Da <u>t</u> einame:	c:\temp\kunden.html	Durchsuchen
			Formatoptionen
		Daten von System i übertragen	

Datenexport

Um die Daten zu exportieren, klicken wir auf die Schaltfläche "Daten von System i übertragen".

Wenn noch keine Verbindung zum System i existiert, müssen Sie sich zunächst mit einem berechtigten Benutzer anmelden. Danach erscheint dann die Ergebnisanzeige des Exports nach HTML:



Erfolgsmeldung



Web-Programmierung	5
UTML und System i	

HTML und System i

5.1.1 Seite 11

apite

In der Ergebnisanzeige können wir uns den Stand des Exports mittels der Schaltfläche "Datei öffnen" anschauen. Damit verbunden ist der Aufruf zur Darstellung des HTML-Exports in einem Browser. Ein Beispiel dazu sehen Sie in der folgenden Abbildung:

E Kunden - Windows Internet Expl	orer								<u>a</u>	_   <b>  ×</b>
C:\Temp\kunden.html	• ٩	47 🗙	—		<i>е</i> к	lunden	×		6	*
Mittwoch, 5. März 2014 14:07:34					_					<u> </u>
			Aus	gabe Datei KUN	DE in W	ebseite			1	
	KUMANDT	KUKDNR	KUNAME	KUSTRASSE	KUPLZ	KUORT	KUSTATUS	KUUUSER		
	001	2	Willi Winzig	Hauptstrasse 1	57071	Siegen	A			
	001	10	Helmut Kohl	Dorfstrasse	60000	Ockersdorf	A			
	001	15	Heidi Klum	Bergstrasse	54880	Bergisch Gladbach	A			
	001	20	Otto Kabel	Mainstrasse 7	50001	Köln	A			
	002	2	Willi Winzig	Poststrasse 18	57482	Olpe	A			
	002	90	Siegfried Schmidt	Dorfweg 7	61145	Walddorf	A			

#### Ergebnisanzeige

Wie Sie sehen, wurde der Inhalt unserer Kundenstammdatei gemäß unseren Vorgaben aufbereitet, erfolgreich nach HTML exportiert und kann nun beispielsweise in einem Programmaufruf wiederverwendet werden. Natürlich enthält die HTML-Datei bislang nur Basisangaben, aber wir sind ja auch noch nicht am Ende des Funktionsumfangs angelangt, den System i Access für die HTML-Ausgabe bietet.



HTML und System i

5.1.1 Seite 12 Einige der "erweiterten" Funktionen steuern beispielsweise das Verhalten der HTML-Darstellung bei Seitenumbrüchen. Selbstverständlich können wir auch definieren, wie die Anzeige auf mehrere Seiten verteilt werden kann, falls das aufgrund der Datenmenge erforderlich sein sollte. Die dazu notwendigen Einstellungen finden wir im Abschnitt "Tabelle".

HTML-Details				<u></u>
Zeile	Zelle	) Taballa	Schabl	one
Datei Do	okument	Tabelle	Besci	nnttung
Ausrichtung				
C Links	• <u>M</u>	itte	О <u>R</u> e	chts
Attribute				
✓ Inklusive Spa	ltenüberschrifte	n		
Anzahl Zeilen	: 3	-	Optione	n
Anzahl <u>S</u> palte	n: 2			_
Rahmenbreite	(Pixel): 1	-		
Zellenabstand	d (Pixel): 1			
 		-		
	te (Pixel):			
I <u>T</u> abellenbreite	e: [100	<u> </u>	% des Fen	ster 🔽
ОК	Abbrechen	1 Übern	ehmen	Hilfe

#### Tabellenanzeige

Zunächst lässt sich festlegen, ob in die Tabelle eine Spaltenüberschrift eingebunden werden soll, was natürlich stets sinnvoll ist – ob bereits via Programmierung hinterlegt oder mit dieser Funktion definiert. Als Spaltenüberschrift wird in diesem Fall der Feldname der DB2-Datei verwendet.





Seite 13

HTML und System i

Weiterhin kann mit der Angabe "Anzahl Zeilen" das Erscheinungsbild auch hinsichtlich der maximal auf einer Seite darzustellenden Zeilen angepasst werden. Sollte die maximale Anzahl den hier angegebenen Wert überschreiten, führt das in der HTML-Ausgabe automatisch zu einem Seitenumbruch. Wie dieser Seitenwechsel durchzuführen ist, legen wir mit den folgenden Angaben fest:

Anzahl Zeilen 🔤 ? 🗙						
Wenn die Übertragung mehr Zeilen als angegebene Zeilen pro Tabelle zurückgibt:						
O Verbleibende Zeilen abschneiden						
C Mehrere Tabellen in einem Dokument generieren						
Mehrere verbundene Dokumente generieren						
Wenn die Übertragung weniger Zeilen als angegebene Zeilen pro Tabelle zurückgibt: © Überzählige Zeilen ignorieren © Leerzeilen generieren						
OK Abbrechen						

#### Optionen

Mit den Zeilenoptionen bewirken wir die Erstellung von mehreren verbundenen Dokumenten, was dazu führt, dass zum Beispiel eine Schaltfläche "Weiter" automatisch in die Anzeige integriert wird, dass bei Bedarf Leerzeilen entfernt oder überschüssige Zeilen abgeschnitten werden.







5.1.1 Seite 14 Die folgenden beiden Abbildungen zeigen ein Beispiel für die Erstellung einer HTML-Ausgabe, deren Inhalt aus mehrere Zeilen besteht, die auf unterschiedliche, verbundene Dokumente verteilt sind.



Ergebnis 1. Seite
Web-Programmierung HTML und System i	<b>Rapite</b>
Da wir in unserem Beispiel die maximale Zeilenanzahl mit "2" angegeben haben, werden auch nur zwei Zeilen pro Anzeigeseite dargestellt. Steuerungs- schaltflächen für die verbundenen Dokumente wurden bei der Erstellung der HTML-Ausgabe automatisch eingefügt:	5.1.1 Seite 15

Schaltfläche "Weiter"

Schaltfläche "Zurück"

E Kunden - Windows Internet Explore	er									<u>a _ n ×</u>
🔆 🔄 🗢 📳 C:\Temp\kund0000.html	• • ۹	• × 👘			6	Kunden		<i> K</i> unden	×	🗄 🖈 🕸
Mittwoch, 5, März 2014										 <u>~</u>
14:12:58										
			A	usgabe Datei K	UNDE i	n Webseite				
	KUMANDT	KUKDNR	KUNAME	KUSTRASSE	KUPLZ	KUORT	KUSTATUS	KUUUSER		
	001	15	Heidi Klum	Bergstrasse	54880	Bergisch Gladbach	A	]		
	001	20	Otto Kabel	Mainstrasse 7	50001	Köln	A			
				Zurücl	Weit	er				
					-					
I										

Ergebnis 2. Seite

Wäre eine Subfileprogrammierung doch nur ähnlich schnell durchgeführt!

Wie Sie sehen, lassen sich zu den Basiseinstellungen einige weitere Funktionen für den wirklich einfach anzuwenden HTML-Export des System i Access anwenden. Selbstverständlich sind das nur rudimentäre Lösungen - und in keiner Weise "Highend"-HTML-Anwendungen. Aber den Anspruch sollte man hier natürlich auch nicht haben. Wem die Basisexport-Funktionen nicht genügen, der kann sich HTML-Vorlagen bedienen, die -individuell erstellt - die Grundlage für eine optimierte HTML-Ausgabe bilden.



HTML und System i

### 5.1.1 Seite 16

### 5.1.1.2 HTML-Export unter Verwendung einer HTML-Schablone

Einfache Exportfunktionen, bei denen es weniger um die aufbereitete Darstellung als vielmehr um die Bereitstellung der Daten an sich geht, haben wir nun bereits kennengelernt. Nachfolgend erfahren Sie, wie mit Hilfe von individuellen HTML-Vorlagen die HTML-Exportfunktion von System i Access für die Erstellung von ausgereifteren HTML-Ausgaben eingesetzt werden kann.

HTML-Schablonen können mit einem beliebigen Editor erzeugt werden. Hier eignet sich als ganz einfacher Editor durchaus auch Notepad.

System i-Entwicklern bietet sich für die Erstellung von HTML-Seiten der entsprechende Funktionsbereich innerhalb des Rational Developers (RDP) an. Im RDP können wir neben der klassischen Anwendungsentwicklung auch Webanwendungen und HTML editieren. Die dazugehörigen Werkzeuge finden sich in der Web-Ansicht.

Über die Neuanlage einer Datei können wir ein leeres Dokument erstellen, das in einem Folgeschritt als HTML-Datei definiert wird.

Wechseln Sie zunächst in die Ansicht "Web". Dazu können Sie zum Beispiel den Menüeintrag "Fenster/Ansicht anzeigen" verwenden. Sollte in dieser Auswahl der Eintrag "Web" bereits enthalten sein, so verwenden Sie ihn – anderenfalls klicken Sie auf den Eintrag "Andere" im unteren Bereich der Auflistung und wählen dort den Eintrag "Web" aus.

🕑 Ansicht anzeigen	
Filtertext eingeben	
Java     Java durchsuchen     JavaScript     JavaScript     JavaServer-Faces     JPA     Migration     Modelligesteuerte Lösungen     Modellierung     Operational QVT     Operational QVT     Server     Server     Team     Web     Correct Address     Andere	
	OK. Abbrechen

Ansicht auswählen



HTML und System i

apitel

Seite 17

Mit dieser Auswahl wechselt die Darstellung und wir sehen die Bestandteile der Web-Ansicht.

Um nun eine neue HTML-Datei zu erstellen, wählen wir den Menüeintrag "Datei/Neu/Datei".

🕑 Web - ZEIGRPG/test.html - IBM Rationa	I Developer for Power	r Systems Software
Datei Bearbeiten Quelle Navigieren Suche	n Projekt Ausführen	Seite Fenster Hilfe
Neu	Alt+Umschalttaste+N	🕨 🎲 Webprojekt
Datei öffnen		Statisches Webprojekt
Schließen	Strg+W	🔯 Dynamisches Webprojekt
Alle schließen	Strg+Umschalttaste+W	Projekt
Speichern	Stra+S	Webseite
Speichern unter		2 Webseitenschablone
Alle speichern		(€) Klasse
Zurücksetzen		😯 Schnittstelle
		JSP-Datei
Imbenennen		S CSS-Datei
Aktualisteren		💥 JavaScript-Quellendatei
Zeilenbegrenzer umwandeln in		☐ Ordner
		Datei
E Drucken	Strg+P	Beispiel
Arbeitsbereich wechseln		
Erneut starten		Sugriv
r≥nImportieren		
Exportieren		

Neue Datei erstellen

Nun öffnet sich ein neues Fenster für die Eingabe des Dateinamens. Da RDP in diesem Bereich über Projekte organisiert ist, müssen wir zunächst festlegen, in welchem Projekt die neue Datei anzulegen ist. Sollte noch kein Projekt existieren, können Sie sich ein eigenes nach Belieben definieren und anlegen.



HTML und System i

5.1.1 Seite 18 Im Feld "Dateinamen" geben wir nun den gewünschten Namen der HTML-Datei ein, die uns als Schablone für den Datenexport dienen soll. Ich nenne diese HTML-Schablonendatei "KDVorlage" und versehe sie mit der Endung .html.

🕑 Neue Datei	a _ o ×
Datei	
Neue Dateiressource erstellen	
Stammordner eingeben oder auswählen:	
ZEIGRPG	
<ul> <li>Applettutorial</li> <li>BBZugriff</li> <li>BDBZugriff</li> <li>BDZUgriff</li> <li>BDZUgriff</li></ul>	
Dateina <u>m</u> e: kdvorlage.html	
<u>E</u> rweitert >>	
? Eertig stelle	n Abbrechen

Dateinamen angeben



Mit einem Klick auf "Fertig stellen" gelangen wir in den Eingabebereich neuen Datei. Hier wollen wir nun das Erscheinungsbild unserer HTML-F festlegen. Das kann in beliebiger Form geschehen. Für unser Beispiel bes ken wir uns auf die Angabe eines Titels und einige wenige Inhaltsangab	unserer 5.1.1 Exporte Seite 19 schrän- en:
<html></html>	
<head></head>	
<title> Schablone zur Anzeige der Kundenauflis</title>	tung
<body></body>	
<h1>KDVORLAGE ein Beispiel mit HTML Schablone</h1>	
KDVORLAGE	
Web - ZEIGRPG/KdVorlage.html - IBM Rational Developer for Power Systems Software Datei Bearbeiten Navigieren Suchen Projekt Ausführen Fenster Hilfe	
╡┇╸╔╔╔╺ ╡╗┍╗┍┶╺╺╺	│ 2≱ │ 2° \$° Pobug  i-Projekte
😪 Enter 🕱 🔁 Projek 🗖 🗖 📔 KdVorlage.html 🕱	3 ×2
<pre></pre>	

### HTML-Schablone mit RDP erstellen

Es ist keine Darstellung verfügbar.

] 🕈 😑

Mit dem Eintrag <!-- KDVORLAGE --> geben wir den Punkt an, an dem der Exportprozess die HTML-Informationen dynamisch einfügen soll. Die Änderungen speichern wir und schon steht die HTML-Schablone für den Einsatz mit unserer System i Export-Funktion zur Verfügung.

🔝 Fehler 🕱 🛛 🤻 Server 🔲 Eigenschaften 📄 Schnellbearbeitung 📮 Konsole 1 Fehler, 272 Warnungen, 0 Andere (Filter stimmte mit 101 von 273 Elementen überein)

Ressource

Schreibzugriff

Pfad

Einfügen

1:1

Position Typ

Þ



apite

Web-Programmierung

HTML und System i

Beschreibung

**ء** ا

📀 Fehler (1 Element)



HTML und System i

5.1.1 Seite 20 Merken Sie sich den HTML-Vorlagenamen und auch den Namen des gerade definierten Einfügepunkts. Achten Sie darauf, dass beide case sensitiv sind!

Kehren wir nun wieder zu unserem Exportvorgang im System i Access zurück. Hier verwenden wir erneut die bisherigen Angaben – ändern aber für den Export der Daten gemäß unserer HTML-Schablone die Eintragungen im Bereich "Schablone".

HTML-Details			<u>⊜</u> ?×
Datei Zeile	Dokument Zelle	Tabelle	Beschriftung Schablone
HTML-Sch	ablonendatei verw	enden	
C:\Users\Zeig	tei: j\RPG\KdVorlage.	html	Durchsuchen
Eingebettetes	Schablonenkennze \GE>	eichen:	
		1	
OK	Abbrecher	Oberne	hmen Hilfe

### Schablonenangaben

Zunächst hinterlegen wir den Namen unserer erstellten HTML-Vorlage. Sollten Sie die Vorlage mit RDP erstellt haben, so befindet sich die Datei im Arbeitsbereich innerhalb des entsprechenden Projektordners. Sollte Ihnen diese Angabe zu umständlich sein, können Sie die Datei selbstverständlich beliebig verschieben oder kopieren. Wichtig ist hier, den Dateinamen in der korrekten Schreibweise anzugeben!

Dasselbe gilt für die Angabe des Startpunkts, ab dem System i Access die HTML-Informationen einbinden soll. In unserem Fall ist das die Angabe von <!-- KDVORLAGE -->.



Web-Programmierung	
HTML und System i	
lick auf "OK" und exportieren die	5.1.1
ML-Darstellung der System i-Daten	C 11 01

Wir bestätigen die Angaben mit einem K IBM i-Daten erneut. Das Ergebnis: eine HTML-Darstellung der basierend auf einer HTML-Schablone.

# Seite 21

<section-header><section-header><section-header><section-header></section-header></section-header></section-header></section-header>	C:\Temp\kunden.html	-] + <sub>+</sub> -			nden	<b>A</b> 10-1	dan	( and the		
<section-header><section-header><section-header></section-header></section-header></section-header>				KL	nden	Kun	aen	😂 Schablo	ne zur A	× 🗋
MACAGE GIN BEISPIE INTERFICIENCEArgebreitungen Beische UnterfichenName Kunne Kun		л.			<b></b>	~ • • •				
Ausgabe Datei KUNDE in WebseiteKUMANDTKUKDNRKUNAMEKUSTRASSEKUPLZKUORTKUSTATUSKUUUSER0012Willi WinzigHauptstrasse60000OckersdorfA00115Heidi KlumBergstrasse54880Bergisch GladbachA00120Otto KabelMainstrasse 750001KolnA0022Willi WinzigPoststrasse 1857482OtpeA00290Siegfried SchmidtDorfweg 761145WalddorfA	JKLAGE en	n Beis	spiel mi	t HIN		Schablo	ne			
Ausgabe Datei KUNDE in Webseite         KUMANDT       KUKDNR       KUNAME       KUSTRASSE       KUPLZ       KUORT       KUSTATUS       KUUUSER         001       2       Willi Winzig       Hauptstrasse       57071       Siegen       A         001       10       Helmut Kohl       Dorfstrasse       60000       Ockersdorf       A         001       15       Heidi Klum       Bergstrasse       54880       Bergisch Gladbach       A         001       20       Otto Kabel       Mainstrasse 7       50001       Köln       A         002       2       Willi Winzig       Poststrasse 18       57482       Olpe       A         002       90       Siegfried Schmidt       Dorfweg 7       61145       Walddorf       A										
KUMANDTKUKDNRKUNAMEKUSTRASSEKUPLZKUORTKUSTATUSKUUUSER0012Willi WinzigHauptstrasse57071SiegenA00110Helmut KohlDorfstrasse60000OckersdorfA00115Heidi KlumBergstrasse54880Bergisch GladbachA00120Otto KabelMainstrasse 750001KölnA0022Willi WinzigPoststrasse 1857482OlpeA00290Siegfried SchmidtDorfweg 761145WalddorfA			Aus	gabe Datei KUI	VDE in W	ebseite				
0012Willi WinzigHauptstrasse 157071SiegenA00110Helmut KohlDorfstrasse60000OckersdorfA00115Heidi KlumBergstrasse54880Bergisch GladbachA00120Otto KabelMainstrasse 750001KölnA0022Willi WinzigPoststrasse 185742OlpeA00290Siegfried SchmidtDorfweg 761145WalddorfA	KUMAND	T KUKDNR	KUNAME	KUSTRASSE	KUPLZ	KUORT	KUSTATUS	KUUUSER		
00110Helmut KohlDorfstrasse6000OckersdorfA00115Heidi KlumBergstrasse5480Bergisch GladbachA00120Otto KabelMainstrasse 750001KölnA0022Willi WinzigPoststrasse 1857482OlpeA00290Siegfried SchmidtDorfweg 761145WalddorfA	001	2	Willi Winzig	Hauptstrasse 1	57071	Siegen	A			
00115Heidi KhmBergstrasse5480Bergisch GladbachA00120Otto KabelMainstrasse 750001KölnA0022Willi WinzigPoststrasse 1857482OlpeA00290Siegfried SchmidtDorfweg 761145WalddorfA	001	10	Helmut Kohl	Dorfstrasse	60000	Ockersdorf	A			
001       20       Otto Kabel       Mainstrasse 7       50001       Köln       A         002       2       Willi Winzig       Poststrasse 18       57482       Olpe       A         002       90       Siegfried Schmidt       Dorfweg 7       61145       Walddorf       A	001	15	Heidi Klum	Bergstrasse	54880	Bergisch Gladbach	ı A			
002     2     Willi Winzig     Poststrasse 18     57482     Olpe     A       002     90     Siegfried Schmidt     Dorfweg 7     61145     Walddorf     A	001	20	Otto Kabel	Mainstrasse 7	50001	Köln	A			
002 90 Siegfried Schmidt Dorfweg 7 61145 Walddorf A	002	2	Willi Winzig	Poststrasse 18	57482	Olpe	A			
	002	90	Siegfried Schmidt	Dorfweg 7	61145	Walddorf	A			

### Ergebnisanzeige

Sie können die HTML-Schablone mit allen zur Verfügung stehenden HTML-Komponenten versehen und damit das Erscheinungsbild der Ausgabe entsprechend anpassen.

Die Verwendung einer HTML-Schablone gestattet so zum Beispiel das Einbinden von Unternehmensinformationen, grafischen Komponenten oder schlicht das Erstellen einer perfekt gestylten Webseite, die Ihrem Unternehmen einen gewissen Charakter verleiht. Dass sich hinter der HTML-Erzeugung ein einfaches Werkzeug verbirgt, wie in diesem Fall System i Access, erkennt kein Anwender.

Wieder einmal haben wir ein Beispiel dafür kennengelernt, wie einfach IBM es uns System i-Betreibern macht, gewisse Informationen auch über die Grenzen des Systems hinaus verfügbar zu machen. Mit etwas HTML-Know-how lassen sich im Zusammenspiel mit einer Standardfunktion des Systems hochwertige HTML-Ausgaben erzeugen und zum Beispiel in RPG- oder CL-Aufrufe einbinden.





Web-Programmierung HTML und System i

5.1.1 Seite 22





### 5.1.2 Seite 1

### 5.1.2 Ferne Befehle

Seitdem das System i oder auch dessen Vorgänger mit anderen Systemen verbunden werden kann bzw. in der Lage ist, mit diesen zu kommunizieren, existiert auch die Forderung, Befehle von "fernen Systemen" – hier sind z. B. PC-Anwendungen gemeint – auf dem System i ausführen zu können.

Solche fernen Befehle können wir uns zunutze machen, um damit beispielsweise PC-Programme zu starten, Anwendungen zu initiieren oder auch Webseiten für die Anbindung an RPG-Anwendungen aufzurufen.

Das Ausführen ferner Befehle wird bereits seit frühen OS/400-Releases angeboten und wird auch heute noch mit den aktuellen Versionen des IBM i unterstützt. Man kann in diesem Zusammenhang sicher von "in die Jahre gekommenen Lösungen" sprechen, Lösungen jedoch, die nach genauem Abwägen durchaus in der Lage sind, in modernen Umgebungen Einsatz zu finden.

Mit dem Betriebssystem IBM i haben wir unterschiedliche Optionen, ferne Befehle auf Zielrechnern auszuführen, z.B.:

- Befehl RUNRMTCMD
- Befehl STRPCCMD

Leider, und das muss bereits an dieser Stelle betont werden, gibt es bestimmte Einschränkungen, die den Einsatz von fernen Befehlen schon in der Analysephase ausschließen. Diese Einschränkungen werden wir später betrachten.

Allen Optionen, mit denen wir in der Lage sind, ferne Befehle auszuführen, ist die Grundlage für die Ausführung ferner Befehle gemein. Damit ein ferner Befehl mit IBM i-Mitteln ausgeführt werden kann, müssen auf dem PC bestimmte Voraussetzungen erfüllt sein. Diese werden in der Regel über iSeries Access eingestellt.

Ferne Ausführungen von Befehlen werden allgemein als "Remote Execution" (kurz: REXEC) bezeichnet.

Nicht nur Java-Anwendungen erfordern es, dass auf dem System i Server Anwendungen remote gestartet werden können, sondern auch klassische Anwendungen benötigen z.B. für eine PC-Aktion den Aufruf eines Programms auf dem System i. Beispiele für den Einsatzbereich solcher "fernen Befehle" sind weit verbreitet. Nachfolgend einige Musterbereiche, für die ein ferner Befehl sinnvoll sein kann:

- Ermitteln von Umsatzzahlen, die vom System i in ein PC-Programm (z. B. Excel) übertragen und dort aufbereitet werden.
- Abrufen von Kalkulationen, um sie mit einem PC-Programm weiterzuverarbeiten.



Ferne Befehle

5.1.2 Seite 2

- Im Workflow eines Windows-Rechners der Aufruf eines Batchprogramms, das auf dem System i bestimmte Berechnungen durchführt, die erst dann vorgenommen werden dürfen, wenn auf dem fernen Rechner die notwendigen Aktionen erfolgreich abgeschlossen wurden.
- Nach Abschluss bestimmter Arbeiten auf dem System i Aufruf einer Anwendung auf einem fernen System, welche die zuvor auf dem System i erstellten Daten benötigt.
- Umschalten von Ressourcen auf einem anderen System i.

Bei solchen Interaktionen müssen natürlich die spezifischen Eigenschaften und Anforderungen des Zielsystems (des fernen Systems) bedacht und implementiert werden. Darunter fallen beispielsweise auch die sicherheitsrelevanten Einstellungen, die das Ausführen von Programmen oder Befehlen auf dem System i erst ermöglichen.

Vielleicht ist Ihnen die Ausführung von fernen Befehlen von einem PC aus auf dem System i bereits bekannt: Der Befehl RUNRMTCMD dient u. a. dazu, von einem System aus einen Befehl auf dem System i abzusetzen bzw. auszuführen. Da der Befehl RUNRMTCMD auch in i5/OS zur Verfügung steht, können wir ihn vom System i aus nutzen, um auf einem anderen fernen System einen Befehl aufzurufen. Dabei kann es sich bei dem fernen System um einen beliebigen Rechner handeln, also beispielsweise ein anderes System i, einen Windows-Rechner oder ein UNIX-System.

Basis für diese Funktionen ist ein TCP/IP-Standard, der nicht nur im i5/OS, sondern auch in anderen Betriebssystemen implementiert wurde – der REXEC-Standard. Dieser besondere TCP/IP-Bereich überträgt die von einem Client abgesetzten Befehle auf einen Zielserver, um sie dann dort auszuführen. Eine solche Übertragung besteht aus den folgenden Komponenten:

- Auszuführender Befehl
- Benutzerprofil
- Kennwort für das Benutzerprofil

Als Rückgabe erhält der Client das Ergebnis der Befehlsausführung von dem fernen System. Die nachfolgende Grafik verdeutlicht dieses Verfahren:





Seite 3



REXEC-Ausführung grafisch

In Bezug auf das System i bedeutet die REXEC-Anbindung und -Ausführung, dass auf der Maschine ein spezieller TCP/IP-Serverdienst gestartet werden muss, der die fernen Befehle entgegennimmt.

Die REXEC-Anbindung / -Realisierung mit dem System i bedeutet, dass unterschiedliche Befehlsformen an den Server übertragen werden können. Dies sind die folgenden Bereiche:

- Native System i-Befehle
- QShell-Befehle

### Achtung!

Bei der Verwendung der REXEC-Funktionen zur Übergabe von Befehlen an das System i ist darauf zu achten, dass diese Befehle als Hintergrundjobs auf dem System ausgeführt werden.

Neben der Notwendigkeit, Besonderheiten der Batchjobs, die mit dem REXEC-Bereich des System i verarbeitet werden können, zu beachten, müssen auch Grenzwerte eingehalten werden, wie z. B. die maximale Länge eines Befehls, der von dem REXEC-Prozess verarbeitet werden kann. Dieser ist auf 4.000 Bytes limitiert.

Wenn von einem Client ein Befehl an den REXEC-Server übergeben wird, dann wird dazu der Befehl RUNRMTCMD verwendet. Er arbeitet auf Basis von zwei einzelnen Verbindungen, die auch dazu genutzt werden, um die Rückinformationen entgegenzunehmen. Nachfolgend finden Sie ein einfaches Beispiel für einen solchen Befehl:

```
RunRmtCmd Cmd(,Start E:\MyDirectory\MyProgram.exe')
RmtLocName(1.2.3.4
```

```
*IP) RMTUSER(*None) RMTPWD()
```





Ferne Befehle

5.1.2 Seite 4 Im Falle der Ausführung des Befehls auf dem System i finden wir den Befehl RUNRMTCMD wie folgt vor:



### RUNRMTCMD

Verwendet man beispielsweise einen Windows-Client, dann wird mit dem Befehl RUNRMTCMD automatisch eine Verbindung zur Rückgabe der Fehlermeldungen und eine weitere Verbindung für die Rückgabe der Standardmeldungen genutzt. Diese stehen anschließend auf dem Client zur Auswertung zur Verfügung. Auf diese Weise kann man feststellen, ob der ferne Befehl ordnungsgemäß verarbeitet wurde.

Grundsätzlich gilt beim Einsatz von REXEC, dass ein Client (oder Server) auf einem fernen Rechner einen Befehl absetzen kann. Dabei spielt es keine Rolle, ob es sich bei dem Rechner, von dem der ferne Befehl abgesetzt wird, um einen PC, einen Server oder ein System i handelt. Damit ist dann auch klar, dass der ferne Aufruf auch von einem System i zu einem anderen System i eingesetzt werden kann.

Das schöne bei einem solchen Szenario: Das System i gestattet das Automatisieren von CL-Aufrufen in einem CL-Programm. Da es sich bei dem Aufruf eines fernen Befehls um einen CL-Befehl handelt, kann man diesen auch problemlos in ein CL-Programm einbinden und damit die Verarbeitung automatisieren.





Seite 5

Die Grundlage für das Ausführen eines fernen Befehls ist TCP/IP. Um einen fernen Befehl abzusetzen, müssen wir also zunächst eine TCP/IP-Verbindung zu dem fernen System herstellen. Das ist unabhängig davon notwendig, um welches lokale bzw. auch um welches ferne System es sich im Einzelnen handelt.

Nach dem Herstellen der Verbindung wird zunächst das Benutzerprofil mitsamt dem zugehörigen Kennwort und dem auszuführenden fernen Befehl übertragen. Als Rückgabewert erhalten wir nach der Ausführung des Befehls auf dem fernen System jede dort erstellte Ausgabe. Abschließend wird die Verbindung beendet.

### Der Befehl STRPCO

Damit auf diese Weise ein ferner Befehl abgesetzt werden kann, benötigen wir zunächst den sogenannten "PC Organisator", der über den Befehl STRPCO (Organisator starten) gestartet werden kann. Die folgende Abbildung zeigt den Befehl mit dessen Parameter Text Assist (PCTA):

3 Sitzung A - [24 x 80]	
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hilfe	
Organisator starten (STRPCO)	
Auswahl eingeben und Eingabetaste drücken.	
Text Assist PCTA <u>*YES</u>	
	Endo
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F12=Abbrechen	LINGE
F13=Verwendung der Anzeige F24=Weitere Tasten	
MA MW	05/050
(3)* (1902 - Sitzung wurde errolgreich gestartet.	10

### STRPCO

Der Parameter PCTA steuert, ob die PC-Funktion Text-Assist verwendet werden soll.

In diesem Beispiel gehe ich davon aus, dass es sich bei dem fernen PC um einen Windows PC handelt. Die Ausführung ferner Befehle ist auch für Nicht-Windows-Systeme (z. B. Linux oder Mac) verfügbar.





Ferne Befehle

### 5.1.2 Seite 6

Wie Sie später noch sehen werden, ist die Ausführung des Befehls STRPCO für eine Einheit als Grundlage für die Verwendung des Befehls STRPCCMD (PC-Befehl ausführen) erforderlich.

Achten Sie darauf, dass STRPCO nur einmal pro Sitzung aufgerufen werden darf. Anderenfalls erhält man die Fehlermeldung IWS4010, die besagt, dass STRPCO bereits zuvor für die Einheit gestartet worden ist.

3 <mark>1</mark> Sitzung A - [24 x 80]		
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hilfe		
MAIN System i-Hauptmenü		
	System:	ASZEBIS
Auswahlmoglichkeiten:		
1. Benutzeraufgaben		
2. Büroaufgaben		
3. Allgemeine Systemaufgaben		
<ol><li>Dateien, Bibliotheken und Ordner</li></ol>		
5. Programmierung		
6. Datenfernverarbeitung		
7. System definieren oder ändern		
8. Problembehandlung		
9. Menu anzeigen		
11. System i Access-Aufgaben		
90. Abmelden		
Auswahl oder Befehl		
===> <u>strpco</u>		
F3=Verlassen F4=Bedienerführung F9=Auffinden F12=Abb	brechen	
F23=Anfangsmenu festlegen		
System 1 Access-Organisator für Einheit QPADEV0001 bereits	s aktīv.	
ME A MW		18/052
ල්" [1902 - Sitzung wurde erfolgreich gestartet.		

STRPCO bereits ausgeführt

### Der Befehl STRPCCMD

Als weitere Möglichkeit für das Absetzen ferner Befehle steht uns der Befehl STRPCCMD zur Verfügung. Mit diesem Befehl lassen sich Befehle an einen PC vom System IBM i aus senden. Bei den Befehlen kann es sich beispielsweise um ausführbare Programme oder auch einen DOS-Befehl handeln.



Ferne Befehle

**E** Kapitel

Seite 7

Die nachfolgende Abbildung zeigt exemplarisch die Verwendung des Befehls STRPCCMD:

© Sitzung A - [24 x 80]	
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hilfe	
• • • • • • • • • • • • • • • • • • •	
PC-Befehl sta	arten (STRPCCMD)
Auswahl eingeben und Eingabetaste drüc	cken.
PC-Befehl PCCMD	D > <u>'start http://www.midrangemagaz</u>
in.de'	
Pause PAUSE	E > <u>*NO</u>
	Ende
F3=Verlassen F4=Bedienerf. F5=Aktu	ualisieren F12=Abbrechen
F13=Verwendung der Anzeige F24=Wei	itere Tasten
Mê A MW	05/050
🗊 1902 - Sitzung wurde erfolgreich gestartet.	

### STRPCCMD

Wie Sie sehen, verfügt der Befehl STRPCCMD über lediglich zwei Parameter :

- PCCMD dabei handelt es sich um den auf dem PC auszuführenden Befehl. Dieser Befehl richtet sich nach dem Betriebssystem, das auf dem Ziel-PC installiert ist. In unserem Beispiel verwende ich den Befehl "starT" und füge diesem die URL einer Webseite hinzu. Damit teilen wir einem Windows-System mit, dass der Browser mit der spezifischen URL geöffnet werden soll.
- 2. PAUSE mit diesem Parameter legen Sie fest, ob der PC nach der Ausführung des Befehls pausieren soll, damit der Anwender die Möglichkeit hat, ggf. einzugreifen. So wird quasi die Steuerung des Anwendungsflusses zwischen iSeries Access und der PC-Anwendung geregelt.

Schauen wir uns den Befehl STRPCCMD und die Auswirkung der Parameter anhand eines kleinen Beispiels einmal an.





Ferne Befehle

5.1.2 Seite 8 Führen wir den Befehl wie in der Abbildung zuvor gezeigt aus, so startet der Explorer und die angegebene Webseite wird aufgerufen:



Webseite nach Aufruf STRPCCMD

Ändern wir nun den Parameter PAUSE auf \*YES:



STRPCCMD mit PAUSE YES



Ferne Befehle	
Abweichend zur vorherigen Ausführung wird zusätzlich ein DOS-Fenster geöffnet:	5.1.2 eite 9
🖾 Eingabeaufforderung für PCO-Sitzung A	
Zur <sup>3</sup> ckkehren mit beliebiger Taste_	

### DOS-Fenster

Ergänzung 16/2011

Dieses dient der Flusssteuerung. Parallel zu dem DOS-Fenster wurde auch der Browser mit unserer Webseite wieder geöffnet. Über das DOS-Fenster können bei Bedarf weitere Steuerungen auf der PC-Seite durchgeführt werden.

Diese einfachen Beispiele veranschaulichen die Verwendung des Befehls STRPCCMD.

Wie Sie sehen, ist dieser in der Verwendung recht einfach.

Achten Sie darauf, dass vor dem Ausführen des Befehls STRPCCMD der Befehl STRPCO abgesetzt werden muss. Anderenfalls schlägt die Ausführung des Befehls STRPCCMD fehl. STRPCO muss für die Einheit einmalig ausgeführt werden. Wurde dies nicht gemacht, dann erhalten wir die Fehlermeldung "Befehl zur Zeit nicht erlaubt": apite

Kapitel

### Web-Programmierung

Ferne Befehle

### 5.1.2 Seite 10

31 Sitzune A - [24 x 80]		
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hilfe		
MAIN Sustem i-Hauntmenii		
	Sustem:	ASZEBIS
Auswahlmöglichkeiten:		
1. Benutzeraufgaben		
2. Büroaufgaben		
3. Allgemeine Systemaufgaben		
4. Dateien, Bibliotheken und Ordner		
5. Programmierung		
b. Datenternverarbeitung Z. Sustem dafinianen adam öndern		
7. System derinferen oder andern 2. Droblombobandlung		
9. Monii anzoigon		
5. Hend dilzergen		
11. Sustem i Access-Aufgaben		
90. Abmelden		
Auswahl oder Befehl		
===>		
F3=Verlassen F4=Bedienerfunrung F9=Auffinden F12=A	bbrechen	
P23=HnlangSmenu resilegen Rofobl zur Zoit nicht orlaubt		
		10,000
ML E MW xQ 1002. Site was words set always be set article		167030
Gr (pace - biczung wurde en olgreich gestartet.		

### Befehl nicht erlaubt

Als Einschränkung für den Einsatz des Befehls STRPCCMD ist zu nennen, dass er nur für interaktive Jobs angewendet werden kann. Eine Einbindung in einen Batchjob ist nicht möglich.

Eine weitere Einschränkung ist die Begrenzung der maximalen Zeichenlänge, die mit dem Befehl STRPCCMD übergeben werden kann. Sie liegt bei maximal 123 Zeichen (!) – viel zu kurz, wie Sie vielleicht in einigen Fällen erkennen werden. Die Verwendung von STRPCCMD ist weiterhin auf den Einsatz von iSeries Access (bzw. System i Access) beschränkt. Eine Verwendung zusammen mit anderen Emulationen ist offiziell nicht unterstützt, kann aber natürlich für den verwendeten Emulator auch getestet werden.



Ferne Befehle

## **C** Kapitel

Seite 11

### 5.1.2.1 Ferne Befehle und RPG

Natürlich lassen sich ferne Befehle nicht nur über Befehlseingabe, sondern auch über die Einbindung in Programme (z. B. CL oder RPG) anwenden.

Schauen wir uns einmal an, wie man die Ausführung eines fernen Befehls mit einem RPG-Programm automatisieren kann. Dazu verwende ich in dem Beispiel Freeformat RPG. Dies ist kein Muss, sondern die Einbindung der fernen Befehlsausführung kann auch im Fixformat codiert werden.

Schauen wir uns einmal ein Beispielprogramm an:

D\* RPG-Beispiel für das Absetzen eines fernen Befehls D\* Hier: Aufruf einer Webseite über Internet Explorer und Windows XP D QCMDEXC PR ExtPgm(,QCMDEXC') D\* D cmd 32702A const options(\*varsize) 15P 5 const D len D\* cmd 200A varying S D cmd S 256A varying D URL 256A s 9 0 D Laenge S /free monitor; // Absetzen Befehl STRPCO cmd = ,STRPC0 PCTA(\*N0)'; QCMDEXC(cmd: %len(cmd)); on-error; endmon; // Zusammenbauen des fernen Befehls // In diesem Beispiel starten wir den // Internet Explorer und zeigen eine Webseite an cmd = ,STRPCCMD PCCMD(, 'start HTTP://www.Midrangemagazin. de'') -PAUSE(\*NO)';

Ferne Befehle

```
Laenge = %Len(Cmd);
// Und jetzt wird der ferne Befehl ausgeführt
QCMDEXC(cmd: %len(cmd));
    *inlr = *on;
/end-free
```

In dem Codebeispiel verwende ich QCMDEXC. Mit diesem API lassen sich aus RPG heraus IBM i-Befehle aufrufen.

Um sicherzustellen, dass ein ferner Befehl mit STRPCCMD abgesetzt werden kann, wird in dem RPG-Code jedes Mal der Befehl STRPCO abgesetzt. Dazu dient der folgende Codeabschnitt:

```
cmd = ,STRPCO PCTA(*NO)';
QCMDEXC(cmd: %len(cmd));
```

Um Fehler bei der Ausführung des Befehls abzufangen, wird die Befehlsausführung in einer Monitorgruppe überwacht. Das macht deshalb Sinn, weil bei einem mehrfachen Aufruf von STRPCO die Fehlermeldung IWS4010 ausgegeben wird. Mit der Monitorgruppe werden in dem Beispiel alle auftretenden Fehler überwacht und ignoriert.

Im weiteren Codeverlauf bauen wir den auszuführenden Befehl zusammen.

```
cmd = ,STRPCCMD PCCMD(,'start HTTP://www.Midrangemagazin.
de'') -PAUSE(*NO)';
```

Wie Sie später noch sehen werden, macht das das Programm flexibler, denn der Befehl kann auf diese Weise bei Bedarf aus Variablen zusammengebaut werden.

Diesen Befehl rufen wir aus RPG mit dem API QCMDEXC wieder auf:

QCMDEXC(cmd: %len(cmd));

Das Programm führt die beiden Befehle STRPCO und STRPCCMD nacheinander aus und ruft in dem Beispiel die gewünschte Webseite auf.





### 5.2 Seite 1

**PHP** steht für "**P**HP: **H**ypertext **P**reprocessor" (ursprünglich "**P**ersonal **H**ome Page Tools") und ist eine Programmiersprache mit einer an Java beziehungsweise an Perl angelehnten Syntax, die hauptsächlich zur dynamischen Erstellung von Web-Seiten oder Web-Anwendungen verwendet wird. PHP ist eine Open-Source-Software.

PHP zeichnet sich besonders durch die leichte Erlernbarkeit, die breite Datenbankunterstützung und Internet-Protokolleinbindung sowie die Verfügbarkeit zahlreicher, zusätzlicher Funktionsbibliotheken aus. Es existieren beispielsweise Programmbibliotheken, um Bilder und Grafiken zur Einbindung in Web-Seiten dynamisch zu generieren.

PHP ist eine serverseitig interpretierte Sprache. Das bedeutet, dass der Quelltext – im Gegensatz zu JavaScript, das in der Regel clientseitig ausgeführt wird – nicht an den Browser übermittelt wird, sondern an einen Interpreter auf dem Web-Server. Erst dessen Ausgabe wird an den Browser geschickt. In den meisten Fällen ist dies ein HTML-Dokument, wobei es mit PHP aber auch möglich ist, andere Dateitypen – wie zum Beispiel Bilder oder PDF-Dateien – zu generieren.



### PHP-Funktionsweise

5.2

**Programmieren mit PHP** 

Mit PHP wie auch mit Perl lassen sich Skripte schreiben, die nichts mit dem World Wide Web (WWW) zu tun haben. Die GTK-Erweiterung stellt sogar eine Programmierschnittstelle für eine grafische Oberfläche zur Verfügung, für die weder Web-Server noch Browser benötigt werden. Verglichen mit den Allzweck-Skript-Sprachen Python und Perl sind die Schnittstellen für die grafische Oberfläche und zu anderen Betriebssystemfunktionen eher spärlich und werden kaum verwendet. PHP spielt seine Stärken vor allem beim Einsatz auf Web-Servern aus.

Programmieren mit PHP

### 5.2 Seite 2

Um eine PHP-Datei im Rahmen einer Web-Anwendung ausführen zu können, benötigt man ein System, das mit den in der Datei enthaltenen Anweisungen umgehen kann. Aus diesem Grund wird durch eine Schnittstelle (beispielsweise ISAPI oder CGI) der Interpreter von einem Server-Daemon bzw. Server-Dienst (zum Beispiel Apache oder IIS) ausgeführt. Die Kombination von Linux/ Windows als Betriebssystem, Apache als Web-Server, MySQL-Server als Datenbank und PHP wird LAMP bzw. WAMP genannt. Fertige LAMP- und WAMP-Pakete, die das einzelne Laden und Konfigurieren von Paketen aus dem WWW unnötig machen, werden im Projekt XAMPP entwickelt, wobei es inzwischen Versionen für Linux, Solaris, Windows und Mac OS X gibt.

Seit Anfang 2006 stellt die Firma Zend eine PHP-Distribution für i5/OS zur Verfügung, die die Spezifika des Betriebssystems i5/OS berücksichtigt und native sowie SQL-Datenbank-Zugriffsmethoden, Programmaufrufe mit Parameterübergaben, Zugriffe auf Systemobjekte (zum Beispiel: Data queues, Data areas, Systemwerte, Spool-Ausgaben etc.) zur Verfügung stellt.

Die Vorteile der serverseitigen Ausführung sind, dass beim Client (Browser) keine speziellen Fähigkeiten erforderlich sind oder Inkompatibilitäten auftreten können, wie es beispielsweise bei JavaScript und den verschiedenen Browsern der Fall ist. Außerdem bleibt der PHP-Quelltext der Seite auf dem Server, und nur die generierten Daten sind für den Besucher einsehbar. Gleiches gilt für andere Ressourcen – wie zum Beispiel Datenbanken, die daher auch keine direkte Verbindung zum Client benötigen.



PHP on System i – Zend Core und Zend Platform

Web-Programmierung

### 5.2.1 PHP on System i – Zend Core und Zend Platform

Für V5R3 und V5R4 wird PHP jetzt auch offiziell von IBM für das System i unterstützt. Daraus ergibt sich eine gradlinige und einfache Installation von PHP als Lizenzprogramm.

Zend PHP besteht aus zwei Komponenten:

- 1. Zend Core for i5/OS: Das ist die eigentliche PHP-Laufzeitumgebung. Diese Komponente muss installiert werden und wird mit der Lizenzprogramm-Nummer 1ZCORE5, Komponente \*BASE bezeichnet.
- Zend Platform for IBM i5/OS: Hier handelt es sich um eine Administrationsoberfläche für die PHP-Umgebung. Diese Komponente muss nicht zwingend notwendig installiert werden. Sie wird mit der Lizenzprogrammnummer 1ZCORE5, Komponente 1 bezeichnet. Beabsichtigen Sie das Zend Studio (PHP-Entwicklungsumgebung von Zend) einzusetzen, dann sollte Zend Platform installiert werden.

Der Funktionsumfang von Zend Core für i5/OS erweitert die Funktionalität von PHP hinsichtlich der System i-Architektur erheblich. So ist beispielsweise – neben SQL-Zugriffen auf die Datenbank – ebenfalls ein satzorientierter Zugriff auf Datenbank-Dateien möglich. CL-Befehle können direkt aus einem SQL-Script aufgerufen werden und Programme mit Parameterübergaben (IN, OUT und INOUT) können gleichermaßen aus einem PHP-Script aufgerufen werden.

Sehen Sie sich dazu die entsprechende Seite an - unter:

www.zend.com

### 5.2.1.1 Download

Die unter i5/OS lauffähige PHP-Version wird von der Firma Zend auf Ihrem Download-Server bereitgestellt und muss von Ihnen heruntergeladen werden.

Den Download finden Sie unter diesem Link:

http://www.zend.com/de/products/zend\_core/zend\_core\_for\_ibm

### **5.2.1** Seite 1









PHP on System i - Zend Core und Zend Platform

### 5.2.1 Seite 2



Zend Core-Download

Klicken Sie auf den im oberen Bild bezeichneten Link.

Zend The php Company	Welcome <b>Klaus-Peter Luttkus</b>	Logout My Zend Support 🔻
Company Products	• Partners Developer Zone Zend Network <mark>Store •</mark>	Search 🔀
<u>Z e n d</u> network	Channels	
Channels     >>       HotFixes     >>       Support     >>       Resources     >>       My Account     >>       Feedback     >>	Select a Zend Core Channel/Sub-Channel:	P Help ↓ Download  HotFixes
	i5/os	
Get Zend Core	Zend Core for i5/0S v.1	<u>1</u>
Support	Zend Studio for IS/OS v5.2 Mac OS X Power	
Special Pricing:	Zena Studio for 15/05 v5.2 Linux x86	<u>+</u> №
30% OFF	Zend Statio for 15/05 V3.2 Windows X86	

Zend Download-Bereich

Hier laden Sie die bezeichneten Komponenten herunter.

Auf Ihrem PC finden Sie danach folgende Dateien:

- 1. ZendCoreFori5\_OS-vXXXXX.zip
- 2. ZendPlatform-2.1.2XXXXX-i5os-i5.zip

XXXX bezeichnet die aktuelle Versionsnummer.

Nachdem alle Downloads fertig gestellt sind, wenden Sie sich Ihrem System i zu.



PHP on System i - Zend Core und Zend Platform

### 5.2.1.2 Voraussetzungen i5/OS

Damit Zend Core auf Ihrem System i installiert werden kann, müssen bestimmte Voraussetzungen Software-technisch erfüllt sein.

Die wichtigste Voraussetzung ist, dass Ihr System i unter i5/OS V5R3 oder V5R4 läuft.

Weiterhin müssen folgende Lizenzprogramme installiert sein:

V5R:

Portable App Solutions Environment	33	5722ss1
Qshell	30	5722ss1
System Openness Includes	13	5722ss1
Digital Certificate Manager	34	5722ss1
Cryptographic Access Provider	*base	5722ac3
IBM Portable Utilities For I5/OS	*base	5733sc1
OpenSSH ,OpenSST, Zlib	1	5733sc1
HTTP Server	*base	5722dg1

V5R4:

Portable App Solutions Environment	33	5722ss1
Qshell	30	5722ss1
System Openness Includes	13	5722ss1
Digital Certificate Manager	34	5722ss1
CCA Cryptographic Service Provider	35	5722ss1
IBM Portable Utilities For I5/OS	*base	5733sc1
OpenSSH ,OpenSST, Zlib	1	5733sc1
HTTP Server	*base	5722dg1

Alle diese Lizenzprogramme sind kostenfrei und Bestandteil der IBM i5/OS-Auslieferung.

Ich betone es noch einmal: Diese Lizenzprogramme müssen installiert sein bzw. installiert werden!

### 5.2.1.3 PTFs

Danach spielen Sie auf Ihrer iSeries das aktuelle CUM-PTF ein, aktualisieren alle oben genannten Lizenzprogramme mit den entsprechenden PTFs und legen darüber hinaus das Gruppen-PTF für den HTTP-Server des System i an: SF99114.

Seite 3



PHP on System i – Zend Core und Zend Platform

### 5.2.1 Seite 4

#### 5.2.1.4 Ressourcen erstellen

Damit die Zend Core-Installation gelingt, muss im IFS folgender Pfad erstellt werden bzw. vorhanden sein:

/usr/local

Im Folgenden sehen Sie die Hardcopies für den i5/OS-Befehl MD (Make directory – Verzeichnis erstellen):

Verzeichnis erstellen	(MD)
Auswahl eingeben und Eingabetaste drücken.	
Verzeichnis DIR	/usr
Allgemeine Datenberechtigung DTAAUT Allgemeine Objektberechtigung . OBJAUT + für weitere Werte Objektprotokollierungswert CRTOBJAUD Prüfoptionen für Objekte CRTOBJSCAN Umben./Verb. aufheb. einschr RSTDRNMUNL	*INDIR *INDIR *SYSVAL *PARENT *NO_

#### MD 1

Verzei	chnis erstellen	(MD)
Auswahl eingeben und Eingabetast	e drücken.	
Verzeichnis	DIR	/usr/local
Allgemeine Datenberechtigung Allgemeine Objektberechtigung . + für weitere We Objektprotokollierungswert Prüfoptionen für Objekte Umben./Verb. aufheb. einschr	DTAAUT OBJAUT rte CRTOBJAUD CRTOBJSCAN RSTDRNMUNL	*INDIR *INDIR *SYSVAL *PARENT *NO_

#### *MD 2*

Achten Sie bitte darauf, dass die folgende Installation von Zend Core unter dem gleichen Benutzer durchgeführt wird, der auch die beiden Befehle MD ausgeführt hat.

Die Installation kann nur dann gelingen, wenn der installierende Benutzer Schreibrechte für diesen Pfad hat.



PHP on System i - Zend Core und Zend Platform

### 5.2.1.5 Sicherungsdateien erstellen

Erstellen Sie eine Sicherungsdatei mit dem Befehl CRTSAVF:

	Sicherungsdatei erste	llen (CRTSAVF)
Auswahl eingeben und	Eingabetaste drücken.	
Sicherungsdatei Bibliothek Text 'Beschreibung'	FILE  TEXT	> <u>ZCOREPROD</u> > <u>IRANSFER</u> *BLANK

Sicherungsdatei ZCOREPROD erstellen

Sicherungsd	atei er	stellen	(CRTSAVF)
Auswahl eingeben und Eingabetaste	drücke	n.	
Sicherungsdatei Bibliothek Text 'Beschreibung'	FILE TEXT	>	ZPLATPROD TRANSFER *BLANK

Sicherungsdatei ZPLATPROD erstellen

### 5.2.1.6 Zend Core-Dateien entpacken

Entpacken Sie jetzt die von der Zend Download-Seite heruntergeladenen Dateien

ZendCoreFori5\_OS-vXXXXXXX.zip und ZendPlatform-2.1.2XXXXX-i5os-i5.zip

mit einem Packprogramm – wie z.B. Winzip.

Danach finden Sie im Entpackungsverzeichnis folgende Dateien:

- 1. ZCOREPROD.SAVF: Das Binär-Image der Bibliothek Zend Core
- 2. ZPLATPROD. SAVF: Das Binär-Image von zusätzlichen PHP-Server-Funktionen
- 3. README: Eine Textdatei mit Anweisungen zur Installation und zum ersten Start von PHP auf dem System i

### Achtung!

Um PHP auf dem System i in Betrieb zu nehmen, reicht die Installation von ZCOREPROD.SAVF!





PHP on System i - Zend Core und Zend Platform

5.2.1 Seite 6

### 5.2.1.7 Zend-Objekte zum System i übertragen

Mittels FTP übertragen Sie die beiden Binär-Images (\*.savf) in die entsprechenden Savefiles.

ftp> bin

ftp> cd transfer

ftp> put "O:\PHP\on iSeries\Zendcore V5R4\zcoreprod.savf"

ftp> put "O:\PHP\on iSeries\Zendcore V5R4\zplatprod.savf"

### 5.2.1.8 Installation von Zend Core

Bevor Sie mit der Installation der neuesten Zend Core-Version beginnen, muss gegebenenfalls eine alte Zend Core-Installation vollständig vom System gelöscht werden.

Dazu führen Sie folgende Arbeitschritte durch:

- 1.) Löschen Bibliothek ZENDCORE
- 2.)Löschen Verzeichnis /usr/local/Zend
- 3.) Löschen Verzeichnis /www/zendcore

Dann beginnt die Installation der aktuellen Zend Core- Version. Die Objekte dieses Programms werden bei dieser neuen Version mit dem Befehl RST-LICPGM als Lizenzprogramm 1ZCORE5 installiert.

Lizenzprogramm zurückspei	ch.	(RSTLICPGM)
Auswahl eingeben und Eingabetaste drücken.		
Produkt LICPGM	>	1ZCORE5
Einheit DEV	>	<u>*SAVF</u>
+ für weitere Werte		
Wahlweise zurückzuspeich. Teil OPTION		<u>*BASE</u>
Zurückzuspeichernde Objektart . RSTOBJ		<u>*ALL</u>
Sprache für Lizenzprogramm LNG		<u>*PRIMARY</u>
Ausgabe OUTPUT		<u>*NONE</u>
Release RLS		<u>*FIRST</u>
Release ersetzen REPLACERLS		<u>*ONLY</u>
Sicherungsdatei SAVF	>	ZCOREPROD
Bibliothek	>	<u>Transfer</u>

Lizenzprogramm 1ZCORE5 installieren



**E** 

Seite 7

PHP on System i – Zend Core und Zend Platform

Während der Installation wird folgender Anfangsdialog automatisch angezeigt. Es kann je nach System jedoch einige Minuten dauern, bevor der erste Bildschirm angezeigt wird.

EPOSIS01	_ 🗆 🗙
Datei Bearbeiten Sicht Kommunikation Aktionen Fenster Hilfe	
	ch. (RSTLICPGM,
Welcome to the Zend Core for i5/0S V1R4M1 installation For complete installation instructions, please refer the Zend Core for i5/0S Installation Guide. F3=Exit Enter=Continue	n to
MA a MW	01/002
ن 1902 - Sitzung wurde erfolgreich gestartet.	1.

Beginn der Anfangskonfiguration

Diesen ersten Bildschirm bestätigen Sie mit Datenfreigabe.

PPOSISO1
Datei Bearbeiten Sicht Kommunikation Aktionen Fenster Hilfe
Zend Core for i5/0S Install
Zend Technologies Ltd Early access/ Beta program License
IMPORTANT - READ CAREFULLY THE FOLLOWING TERMS AND CONDITIONS (THESE "TERMS) BEFORE DOWNLOADING / INSTALLING THE SOFTWARE. BY SELECTING THE "I ACCEPT" BUTTON BELOW, OR BY DOWNLOADING / INSTALLING THE SOFTWARE, YOU AGREE TO BE BOUND BY THESE TERMS. IF YOU SELECT "I DO NOT ACCEPT," THE INSTALLATION PROCESS WILL NOT PROCEED. DO NOT SELECT "I ACCEPT," OR INSTALL THE SOFTWARE UNTIL YOU HAVE CAREFULLY READ, UNDERSTOOD, AND AGREED TO THESE TERMS. IF YOU DO NOT WISH TO AGREE TO THESE TERMS, DO NOT DOWNLOAD OR INSTALL THE SOFTWARE.
THESE TERMS CONSTITUTE THE COMPLETE AND EXCLUSIVE STATEMENT OF THE AGREEMENT BETWEEN YOU, OR THE COMPANY ON WHOSE BEHALF YOU ARE INSTALLING THE SOFTWARE ("YOU"), AND ZEND TECHNOLOGIES, INC. ("ZEND"), WHICH SUPERSEDES ANY PROPOSAL OR PRIOR AGREEMENT, ORAL OR WRITTEN, AND ANY OTHER COMMUNICATION RELATING TO THE SUBJECT MATTER OF THESE TERMS.
WHEN YOU DOWNLOAD THE SOFTWARE, YOUR DETAILS ARE RECORDED, THE EVALUATION PERIOD SHALL COMMENCE AND YOU ARE BOUND BY THE TERMS OF Weitere F3=Decline(NO) Enter=Accept(YES)
ME a MW 01/00
(3)* [1902 - Sitzung wurde erholgreich gestartet.

Lizenzrechte bestätigen

Auch diesen Bildschirm bestätigen Sie mit Datenfreigabe.



5.2.1

Seite 8

Im folgenden Bildschirm geben Sie dem Zend Core-Administrator ein Kennwort.



Administratorkennwort vergeben

Merken Sie sich dieses Kennwort, denn es ist wichtig! Ohne dieses Kennwort können Sie später nicht das Administrationsprogramm für Zend Core und Zend Platform ausführen.

Bestätigen Sie diesen Bildschirm mit Datenfreigabe.

Wenn Sie registrierter Zend-Kunde sind, dann können Sie jetzt via Internet Zend Core aktualisieren lassen.

	EPOSISO1	_ 🗆 🗙
Date	er gearbeiten sjort Kommunikation Heitoren Fenster Fille	
Ι.		
	Zend Core for i5/0S Install	
	You can enter your Zend Core for i5/OS environment over the Internet. To enable this, you must enter Zend Network User ID and Password.	
	If you are not subscribed to Zend Network, please register at: http://www.zend.com/network	
	To enter your Zend Network User ID and Password, select Yes. To continue the installation without entering a Zend Network User ID and Password, select No.	
	F3=N0(Exit) Enter=YES(Continue)	
MĤ	a MW (	067005
-5 <sup>1</sup>	1902 - Sitzung wurde erfolgreich gestartet.	

Registrierter Kunde?

Als registrierter Kunde habe ich den vorherigen Dialog mit Datenfreigabe = YES bestätigt. Nun werde ich zur Angabe meiner Zend-Identifikation aufgefordert.

Wenn Sie den vorherigen Dialog mit F3=NO verlassen haben, so erscheint der folgende Bildschirm nicht.

Zend Core for i5/0S Install Please enter your Zend Network ID (case sensitive). User ID : luttkus@eposgmbh.com Password: F3=Exit Enter=Continue	C EPOSIS01	×
Zend Core for i5/0S Install Please enter your Zend Network ID (case sensitive). User ID : luttkus@eposgmbh.com Password: F3=Exit Enter=Continue		
Zend Core for i5/0S Install Please enter your Zend Network ID (case sensitive). User ID : luttkus@eposgmbh.com Password: F3=Exit Enter=Continue		
Please enter your Zend Network ID (case sensitive). User ID : luttkus@eposgmbh.com Password: F3=Exit Enter=Continue	Zend Core for i5/0S Install	
F3=Exit Enter=Continue	Please enter your Zend Network ID (case sensitive). User ID : <u>luttkus@eposgmbh.com</u> Password:	
	F3=Exit Enter=Continue	
M <mark>A MW 12/034</mark>	M <mark>A M</mark> W	12/034

Zend-ID angeben

Danach ist die Anfangskonfiguration beendet.

Zend Core for i	5/0S V1R4M1 installation completed
For more inform	ation, please go to http://www.zend.com/
Use the Zend Co	re Setup Tool run command: GO ZENDCORE/ZCMENU

Konfiguration erfolgreich beendet!

Mit Hilfe des Menüs ZENDCORE/ZCMENU wird Zend Core administriert.

Auf einer Befehlszeile rufen Sie jetzt das Administrationsmenü auf:

```
GO ZENDCORE/ZCMENU
```



apite

apite

PHP on System i – Zend Core und Zend Platform



Service Management

Wählen Sie die Option "5. Service Management menu".

P EPOSI501		_ 🗆 🗙
Datei Bearbeiten Sicht Kommunikation Aktionen Fenster Hilfe		
ZCAMENU Zend Core for IBM i5/OS Setup Tool		
	System:	EPOSI501
Select one of the following:		
1. Start Zend Core Subsystem 2. Stop Zend Core Subsystem		
4. Start Anache server instances		
5. Stop Apache server instances		
6. ReStart Apache server instances		
8. Start 15_COMD service		
9. Elid 15_comb service		
Selection or command		
===>		
F3=Exit F4=Prompt F9=Retrieve F12=Cancel F23=WRKUSF	RJOB	
		217007
1992 - Sitzung wurde erfolgreich gestartet.		217007

Das Menü ZCAMENU



PHP on System i – Zend Core und Zend Platform

### 5.2.1.9 Jobs im i5/OS

Nach dem Start von Zend Core sollten folgende Jobs im Subsystem QHTTPSVR gestartet sein:

	مر د 1 د				]
	AKTUEITEr				
Subsystem/Job	Benutzer	Art	CPU %	Funktion	Status
ZENDCORE	QTMHHTTP	BCH	0,0	PGM-QZHBMAIN	SIGW
ZENDCORE	QTMHHTTP	BCI	0,0	PGM-QZSRHTTP	SIGW
ZENDCORE	QTMHHTTP	BCI	0,0	PGM-QZSRHTTP	DEQW

Nach dem Start von Zend Core sollten folgende Jobs im Subsystem ZEND gestartet sein:

	Aktueller				
Subsystem/Job	Benutzer	Art	CPU %	Funktion	Status
I5_COMD	QTCP	ASJ	0,0	PGM-EASYCOMD	TIMW
ZC_STR_PRN	ZENDADMIN	BCI	0,0	PGM-prngd	SELW
ZENDCOREAP	QSECOFR	BCI	0,0	PGM-httpd	SELW
ZENDCOREAP	QSECOFR	BCI	0,0	PGM-httpd	DEQW
ZENDCOREAP	NOBODY	BCI	0,0	PGM-httpd	SELW
ZENDCOREAP	NOBODY	BCI	0,0	PGM-httpd	TIMW
ZENDCOREAP	NOBODY	BCI	0,0	PGM-httpd	TIMW
ZENDCOREAP	NOBODY	BCI	0,0	PGM-httpd	TIMW

### 5.2.1.10 Funktionstest

Um die Installation zu verifizieren, starten wir jetzt den Zend Core-Administrator. Geben Sie dazu folgende URL ein:

http://<System i Adresse>:89/ZendCore

Folgender Dialog sollte erscheinen:



Zend Core-Administrator-Login

Seite 11

Grundwerk 4/2007



PHP on System i - Zend Core und Zend Platform

5.2.1 Seite 12 Wenn Sie sich jetzt mit dem Kennwort anmelden, das Sie während der Zend Core-Installation vergeben haben, kommen Sie in einige Server-Administrationsdialoge, auf die ich hier nicht näher eingehen möchte.

Navigieren Sie sich im Control Center zu PHPinfo.



phpinfo

Damit ist die Installation verifiziert.

### 5.2.1.11 Die Zend Core-Umgebung

Speichern Sie alle php-Scripte und HTML-Dokumente in folgendem Pfad:

/www/Zendcore/htdocs



PHP on System i – Zend Core und Zend Platform

### 5.2.1.12 Zend Platform installieren

Dann beginnt gegebenenfalls die Installation der aktuellen Zend Platform-Version. Die Objekte dieses Programms werden bei dieser neuen Version mit dem Befehl RSTLICPGM als Lizenzprogramm 1ZCORE5, Komponente 1 installiert.

Lizenzprogramm zurückspeich. (RSTLICPGM)						
Auswahl eingeben und Eingabetaste drücken.						
Produkt	Zeichenwert Name, *SAVF					
Wahlweise zurückzuspeich. Teil       >         Zurückzuspeichernde Objektart       .         Sprache für Lizenzprogramm       .         Ausgabe       .         Release       .         Sicherungsdatei       .         Bibliothek       .         Bibliothek       .	*BASE, 1, 2, 3, 4, 5, 6, 7 *ALL, *PGM, *LNG Zeichenwert, *PRIMARY *NONE, *PRINT Zeichenwert, *FIRST Zeichenwert, *ONLY, *NO Name Name, *LIBL, *CURLIB					
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F12=Abbrechen F13=Verwendung der Anzeige F24	Weitere F10=Zusätzl. Parameter =Weitere Tasten					

RSTLICPGM: 1ZCORE5, Komponente 1



apite



**Web-Programmierung** PHP on System i – Zend Core und Zend Platform

5.2.1 Seite 14


PHP – Grundsyntax

# 5.2.2 PHP – Grundsyntax

Bevor Sie sich mit PHP auf dem System i beschäftigen, sollten Sie Grundwissen in folgenden Bereichen haben:

- Kenntnis über die Funktionsweise eines Web-Servers
- Grundkenntnisse in der HTML-Syntax (Lesen Sie gegebenenfalls Kapitel 5.3!)
- Professionelle Erfahrung in der Anwendungsentwicklung in einer beliebigen Programmiersprache

Außerdem muss ich Ihnen mitteilen, dass es sich hier nicht um ein komplettes PHP Tutorial handelt, sondern ich konzentriere mich hier auf den Aspekt der Erstellung von kaufmännischen Anwendungen unter PHP, die mit Ressourcen des Systems i unter i5/OS arbeiten.

Wenn Sie eine Gesamtübersicht über die PHP-Syntax benötigen, dann gehen Sie bitte auf folgenden Link:

http://www.php.net/manual/de/

Wenn Sie ein deutschsprachiges Tutorial benötigen, folgen Sie bitte diesem Link:

http://tut.php-q.net/

#### 5.2.2.1 Ein erster Schritt

Grundwerk 4/2007

Um Ihnen einen leichten Einstieg zu ermöglichen, werde ich jedoch zuerst einmal einige Basisaspekte von PHP ansprechen. Wenn Sie bereits Erfahrung in der Programmierung mit PHP besitzen, dann können Sie diesen Abschnitt überspringen.

In den Hardcopies werden Sie sehen, dass ich die PHP-Entwicklungsarbeit mit Hilfe des Zend Studios für i5/OS erledige. Zend Studio für i5/OS können Sie als Beta-Version kostenfrei, jedoch voll funktionsfähig von der Zend Homepage (*www.zend.com*) herunterladen. Es ist jedoch nicht zwingend notwendig, dass Sie mit diesem Werkzeug arbeiten! Selbst ein einfacher Texteditor wie Windows Notepad sollte für den Anfang reichen.

Für WDSC existiert derzeit noch kein gutes PHP Plugin. Arbeiten Sie mit der nativen Eclipse Workbench ab Version 3.2, so können Sie für dieses Tool ein PHP Plugin von der Eclipse-Homepage (*www.eclipse.org*) herunterladen.













Seite 2

Web-Programmierung

PHP – Grundsyntax

Dieses Plugin befindet sich noch in der Entwicklung und ist momentan bezüglich seiner Funktionen für eine professionelle Anwendungsentwicklung mit PHP nicht wirklich brauchbar.

PHP ist eine Script-Sprache. PHP-Scripte werden innerhalb des Apache Web-Servers durch den PHP-Präprozessor abgearbeitet. Die vom Präprozessor erzeugte Ausgabe ist dann HTML-Code, der an den Browser gesendet wird.

Sehen Sie sich bitte folgendes kleines Script an:



Das erste Script: PHP01.php

In diesem Beispiel wird die als Überschrift formatierte Zeichenfolge "Mein erstes Script!" mit der Anweisung "echo" an den Browser gesendet.

Hier sehen Sie, wie Sie PHP-Code in ein Script einfügen. Damit der Präprozessor PHP erkennt, kennzeichnen Sie den Anfang einer PHP-Anweisung oder eines ganzen Anweisungsbereich durch das Anfangstag "<?php" und beenden den PHP-Bereich durch "?>". Innerhalb des PHP-Bereichs darf kein HTML auftauchen. Wenn ja, dann nur im Rahmen einer PHP-Ausgabefunktion wie "echo". Achten Sie auf das Tag "<h1>" beziehungsweise "</h1>".

#### Achtung!

Eine PHP-Anweisung wird mit einem Semikolon beendet.

Die PHP-Syntax ist vollkommen formatfrei und kann nach eigenem Ermessen eingegeben werden.

Sie sehen in diesem ersten Beispiel, dass HTML- und PHP-Code in einem Script gemeinsam verwendet werden können. Während HTML-Ausgaben innerhalb des "<body>"-Tags erfolgen müssen, kann der PHP-Code beliebig platziert werden.



	PHP – C
e Ausgabe dieses Scripts sieht dann wie folgt aus:	
PHP01 - Mozilla Firefox	
atei <u>B</u> earbeiten <u>A</u> nsicht <u>C</u> hronik <u>L</u> esezeichen E <u>x</u> tras <u>H</u> ilfe	0
🗧 🔹 🛶 👻 📀 🏠 🗋 http://172.16.110.22/SWE 💌 🕨 💽 🕻 Google	Q
DHD DH HD EDOS DE EDOS Intern D DoMuŭdmin IIM iSeries úccess für W	
	"
PHP01  Mein erstes Script!	*
PHP01  Mein erstes Script!	*
PHP01 SILLEN SCRIPT!	*
PHP01 Control of the cost of the second with the physical second with the physical second with the physical second second with the physical second se	*
PHP01 Stript!	*
PHP01 Control of the cross for with the physical state	*

PHP01.php - Ausgabe

Wichtig! Die Ausführung eines PHP-Scripts ist beispielsweise nicht mit der Ausführung eines RPG-, COBOL- oder CL-Dialogprogramms vergleichbar. Ein RPG-Programm bleibt bei einem EXFMT, das COBOL-Programm bei einer WRITE/READ-Kombination und das CL-Programm bleibt bei einem SNDRCVF stehen. Ein PHP-Script wird immer komplett ausgeführt!

Wir erweitern unsere Ausgabe. Es soll wieder eine Überschrift ausgegeben werden:

Mein zweites Script!

Dann, getrennt durch einen Absatz:

Mein Trainer sagte: "Das ist aber toll!"

apitel

Seite 3

Web-Programmierung



PHP – Grundsyntax

5.2.2 Seite 4 So sieht das Script dann aus:

	editor 🚱 Browser
1 [	3 <html></html>
2	<head></head>
3	<pre><meta content="text/html; charset=utf-8" http-equiv="Content-Type"/></pre>
4	<title>PHP02</title>
5	
6	
7	
8	php</td
9	
10	echo " <hl>Mein zweites Script!</hl> ";
11	echo "";
12	echo "Mein Trainer sagte:\"Das ist aber toll!\""
13	
14	
15	2>
16 [	-
17	
18	<body></body>
19	
20	
21	

Das zweite Script: PHP02.php

Beachten Sie bitte die Zeichenfolge "Das ist aber toll!". Hier kommt es zu einem Konflikt. Die Anführungszeichen (") sind einmal Zeichenfolge-Begrenzer für die gesamte Zeichenfolge, die die Anweisung "echo" ausgeben soll. In diesem Fall ist das Anführungszeichen (") ebenfalls als Zitatzeichen verwendet worden.

Damit PHP das Anführungszeichen (") nicht als Zeichenfolge-Begrenzer versteht, sondern als reales Zeichen, dass im Browser angezeigt werden soll, muss im Falle des Zitats das Anführungszeichen (") excaped (PHP-Jargon) werden, indem ich vor dieses Zeichen ein Backslash (\) stelle.

Sehen Sie sich jetzt die Ausgabe an:

🕹 PHP02 - Mozilla Firefox	
Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe	$\bigcirc$
	٩
PHP 📑 HP EPOS 🗁 EPOS Intern 📄 PhpMyAdmin 🏧 iSeries Access für W	»
PHP02	•
Mein zweites Script! Mein Trainer sagte:"Das ist aber toll!"	
Fertig	11.

PHP02.php – Ausgabe



	PHP – C
Jeb	en der Ausgabe-Anweisung gibt es noch die Funktion "Print", die ebenfalls
ne	
	r Editor 🥝 Browser
1	<pre>3<html></html></pre>
2	<head></head>
3	<pre><meta content="text/html; charset=utf-8" http-equiv="Content-Type"/></pre>
4	<title>PHP03</title>
5	
6	
7	
8	< 2pnp
9	print(" <bl>Mein drittes Scriptl</bl> ").
11	<pre>print((<nishein );="" drivees="" nis="" print("<n="" seripe.<="">");</nishein></pre>
12	print("Mein Trainer sagte:\"Das ist aber toll!\"");
13	
14	
15	2>
16	3
17	
18	<body></body>
19	
20	
21	

Seite 5

Das dritte Script: PHP03.php

Die Ausgabe ist identisch mit der von PHP02.php.



PHP03.php – Ausgabe

Grundwerk 4/2007

PHP verfügt über eine ganze Reihe an eingebauten Funktionen, von denen Sie einige im Laufe der Zeit kennenlernen werden.

Beenden wir diese kurze Einführung mit dem Aufruf der Funktion phpinfo(). Diese Funktion zeigt uns im Browser Umgebungsdaten unserer PHP-Umgebung an. Wir finden in der Anzeige Konfigurationswerte, die bei der Installation von PHP erzeugt beziehungsweise gesetzt wurden und die in etwa mit den Systemwerten der i5/OS-Umgebung vergleichbar sind.





PHP – Grundsyntax

5.2.2 Seite 6 So sieht das Script aus:



Das vierte Script: PHP04.php

Und hier kommt die Ausgabe:

Bearb	eiten <u>A</u> nsicht <u>C</u> h	rronik Lesezeichen Extras Hilfe		
• 🔿	· · 🕑 🛞 (	1   http://172.16.110.22/5WBuch/PHP04.php	🔹 🕨 💽 🕻 Google	
-P A	HP EPOS 🦳 EPOS I	Intern 📄 PhoMyAdmin 🍱 iSeries Access für W 📄 www.midrange	emaga 🦻 Javascript	
нроз				
	PHP Versi	on 5.1.6	php	
	System	OS400 EPOSI501 4 5 006500091BAC		
	Build Date	Nov 22 2006 07:57:30		
	Configure Command	'/configure' 'prefix=/usr/local/Zend/Core' 'with-config-file-path=// 'enable-force-cgi-redirect' 'enable-fastcgi' 'disable-debug' 'e 'enable-memory-limit' 'disable-all' 'enable-ctype' 'enable-do 'with-libxml-dir=/usr/local/Zend/Core' 'with-openssl=/usr' 'with 'enable-session' 'enable-simplexml' 'enable-spl' 'enable-wo 'with-zib=/usr' 'with-pear' 'with-aps:2=/usr/local/Zend/apache 'with-layout=GNU' 'enable-zmail'	etc' nable-inline-optimization' m''-enable-libxml' -pcre-regex' Jdx' 'enable-xml' 2/bin/apxs'	
	Server API	Apache 2.0 Handler		
	Virtual Directory Support	disabled		
	Configuration File (php.ini) Path	/usr/local/Zend/core/etc/php.ini		
	PHP API	20041225		
	PHP Extension	20050922		
	Zend Extension	220051025		

*PHP04.php – Ausgabe* 

Sollten Sie Probleme mit Ihrer PHP-Umgebung haben, so werden Sie in der phpinfo-Anzeige möglicherweise Daten finden, die Ihnen bei der Analyse der PHP-Umgebung helfen.



**Ein ITP Handbuch** 



- 0 ×

Q

>>

🔻 🕨 🔽 Google

PHP – Grundsyntax

#### 5.2.2.2 Form und Stil

Bezüglich der Programmierung mit PHP gibt zwei Lager. Das eine Lager fordert: In einem PHP-Script soll nur PHP-Code stehen. Welche Auswirkung hat diese Forderung?

Wir nehmen einfach unser erstes Script (PHP01.php) und kommen dieser Forderung nach.

|| 📝 Editor Browser <?php print("<html>"); print("<head>");
print("<head>");
print("<meta http-equiv=\"Content-Type\" content=\"text/html; charset=iso-8859-1\" />"); 4567 print("<title>PHPO1</title>"); print("</head>"); 8 9 10 print("<hl>Mein fauuml:nftes Script!</hl>"); 11 print("</body>"); print("</html>"); 12 13 2>

Lesezeichen Extras Hilfe

📄 PHP 📋 HP EPOS 🛅 EPOS Intern 📄 PhpMyAdmin 🏧 iSeries Access für W... 📄 www.midrangemaga...

http://172.16.110.22/SWBuch/PHP05.php

Das fünfte Script: PHP05.php

Ich speichere es als PHP05.php.

⊆hronik

111

Die Ausgabe:

Datei

PHP01

🐸 PHP01 - Mozilla Firefox

Bearbeiten Ansicht

Mein fünftes Script!

PHP05.php – Ausgabe

Vorteil dieser Variante: Alles ist aus einem Guss!

Mögliche Nachteile: Die Abarbeitung des Scripts ist etwas langsamer, da ja alle Elemente des Scripts durch den PHP-Präprozessor abgearbeitet werden müssen. Sehen Sie sich zum Beispiel das meta-Tag an. Hier musste ich einiges 'escapen', was für mich Mehrarbeit bedeutet.

Soweit zum einen Lager. Das andere sagt: Nur die Elemente in einem Dokument, die als variabel zu betrachten sind und/oder Prozesslogik beinhalten, sind mit PHP abzudecken. Alles andere bleibt reines HTML.









PHP – Grundsyntax

5.2.2 Seite 8 Nehmen wir uns das Script PHP03.php vor und ändern es entsprechend dieser Forderung.

	Browser
1	E <html></html>
2	<head></head>
3	<meta content="text/html; charset=utf-8" http-equiv="Content-Type"/>
4	<title>PHP03</title>
5	
6	
7	
8	<pre><?php print("<hl>Mein drittes Script!"); ?&gt;</pre>
9	
10	<pre><?php print("Mein Trainer sagte:\"Das ist aber toll!\""); ?></pre>
111	
12	<pre>cpodA&gt;</pre>
13	
14	4
15	<pre>//tml&gt;</pre>

Das sechste Script: PHP06.php

Ich speichere dieses Script unter dem Namen PHP06.php ab.

Vorteil dieser Herangehensweise: Der PHP-Präprozessor wird entlastet, womit eine schnellere Anzeige im Browser gewährleistet wird. Außerdem ergibt sich weniger Tipparbeit für den Programmierer.

Nachteile

Der Script-Text wird unübersichtlicher bei ständigem Wechsel zwischen PHP und HTML.

Sie entscheiden für sich, welchem Lager Sie sich zurechnen wollen. Ich gebe hier jedenfalls keine Empfehlung, da diesesThema in der PHP Community sehr kontrovers diskutiert wird.

#### 5.2.2.3 Datenbankzugriffe mit PHP

Um auf die Datenbank des Systems i zuzugreifen, gibt es mittlerweile mehrere Möglichkeiten, die jedoch abhängig von der i5/OS-Version und der verwendeten PHP-Distribution sind.

Der Weg, den ich Ihnen im Folgenden zeigen werde, führt über die ODBC-Funktionen des PHP und gilt für alle Installationsvarianten. Jetzt werden Sie vielleicht sagen, dass ODBC gerade bei größeren Datenmengen durchaus problematisch ist. Aber nicht in dieser Umgebung!

Da PHP und die Datenbank auf ein und demselben System laufen, wird der ODBC-Zugriff nicht über einen eigenen Treiber realisiert, sondern der SQL-Zugriff wird direkt an das SQL-Call Level Interface geleitet, so dass SQL-seitig der Zugriff sehr schnell ist und nicht unter den typischen ODBC-Merkmalen leidet.



PHP – Grundsyntax

# ,

In der Bibliothek QIWS ist eine Kundendatei QCUSTCDT gespeichert. Sofern Sie das Lizenzprogramm iSeries Access für Windows (5722XE1 / 5722XW1) auf Ihrem System i installiert haben, müssten Sie diese Datei auch auf Ihrem System haben.

In dieser ersten kleinen Aufgabe sollen alle Sätze der Datei ausgelesen werden und im Browser in Form einer HTML-Tabelle angezeigt werden. Der Name des Scripts lautet: ODBC01.php.

0

6 8

3

6

2

5 0

4 0

1

6 2

6

0

2

13

Kundenummer

Vornameninitialen

Nachname

Straße

Wohnort

Bundesland

Kreditcode

Endsumme

**Postleitzahl** 

Kreditgrenze

Offene Posten

Aufbau der Datei:

ZONED

CHAR CHAR

CHAR

CHAR

CHAR

ZONED

ZONED

ZONED

ZONED

ZONED

CUSNUM

LSTNAM

STREET

INIT

CITY

STATE

ZIPCOD

CDTLMT

CHGCOD

BALDUE

CDTDUE

5.2.2.3.1.1
PHP-Syntaxelemente

Um unser Script realisieren zu können, benötigen wir einige grundsätzliche Bausteine, die Sie bisher noch nicht verwendet haben bzw. noch nicht kennen.

#### 5.2.2.3.1.2 Variablen

Wir werden, um die Aufgabe fertigzustellen, in unserem Script mit Variablen arbeiten müssen.

Variablen können Strings und Zahlen enthalten, aber auch exotische Daten wie Links, Zugriffshandler und Felder von Variablen (Arrays).

Variablen beginnen in PHP mit dem Dollarzeichen (\$). Danach kommt der Name der Variable, wobei dieser mit einem Unterstrich (\_) oder mit einem Buchstaben anfangen muss, damit PHP diese erkennt. Nach dem ersten Zeichen sind dann fast alle Zeichen erlaubt. Die deutschen Umlautbuchstaben "äöü" sowie "ß" sind im Variablenname erlaubt, doch werden sie von den meist englischsprachigen PHP-Editoren nicht erkannt. Deswegen sollte man sie vermeiden.



# 5.2.2.3.1 Die Aufgabe



PHP – Grundsyntax

5.2.2 Seite 10 Bei Variablennamen wird Groß-/Kleinschreibung unterschieden! Das heißt, die Variable "\$ID" ist eine andere als die Variable "\$Id", "\$iD" oder "\$id".

Eine Typdeklaration existiert in PHP nicht.

Eine Deklaration von Variablen erfolgt über die Wertzuweisung und stellt eine implizite Typdeklaration dar.

#### Arrays (Feldgruppen/Tabellen)

Array sind in PHP sehr wichtig. Viele PHP-Funktionen stellen Daten in tabellarischer Form zur Verfügung beziehungsweise speichern sie in dieser Form.

#### **Erstellen eines Arrays**

Mit der Funktion array können wir ein Array erstellen und es in eine Variable schreiben. Als Parameter der Funktion "array()" geben wir die Werte an, die wir in dem Array speichern wollen. Jeder Wert ist ein Parameter. Die Funktion array kann also beliebig viele Parameter haben. Wenn wir unsere Namen darin speichern wollen, müssen wir das wie folgt tun:

```
<?php
$daten = array("Hans", "Klaus", "Max", "Heinz");
?>
```

Aufdieeinzelnen Elementedes Arraysgreifen wirmit "\$variablenname[nummer]" zu. Für "nummer" setzen wir eine Zahl von 0-3 ein. Wir setzen keine Zahl von 1-4 ein. Denn PHP sowie viele andere Programmiersprachen fangen bei 0 an zu zählen. Da dieses Array, das wir erstellt haben, vier Elemente hat, müssen wir also von 0-3 zählen. Anders ausgedrückt: Wir müssen von 0 bis zur Anzahl der Elemente -1 zählen.

In unserem Beispiel greifen wir auf die Elemente wie folgt zu:

```
<?php
$daten = array("Hans", "Klaus", "Max", "Heinz");
echo "Das 1. Element hat den Wert ".$daten[0]."<br />\n";
echo "Das 3. Element hat den Wert ".$daten[2]."<br />\n";
```

?>



#### Web-Programmierung PHP – Grundsyntax



#### 5.2.2.3.1.3 Wertzuweisungen

Wenn man einen Wert in eine Variable speichern möchte, benutzt man den Zuweisungsoperator "Gleichheitszeichen" (=). Dabei wird der rechte Ausdruck von Ist-Gleich (=) in die Variable, die links steht, hineingeschrieben. Dies könnte zum Beispiel wie folgt aussehen:

```
<?php
   $email = "luttkus@eposgmbh.com";
   echo "Meine Emailadresse ist: ";
   echo $email;
?>
```

Damit wird die erwartete Ausgabe erzeugt:

Meine Emailadresse ist: luttkus@eposgmbh.com

#### Verkettungen

Manchmal möchte man zwei Strings zusammenfügen. Dazu benutzt man den Verkettungsoperator "Punkt" (.). Dabei werden die Teile links und rechts vom Verkettungsoperator zusammengehängt. Dieses "Ergebnis" sollte man dann mit dem Zuweisungsoperator einer Variable zuweisen oder mit "echo" ausgeben.

```
<?php

$variable = "Ein String"." Ein anderer String";

echo $variable;
```

?>

Damit wird die erwartete Ausgabe erzeugt:

Ein String Ein anderer String

#### **Numerische Werte**

In PHP gibt es zwei numerische Datentypen. Einmal sind das Integer-Zahlen und einmal die Float- bzw. Double-Zahlen.

Integer-Zahlen sind ganze Zahlen ohne Nachkommastellen – also zum Beispiel 5, 7 oder 134, aber auch negative Zahlen wie -345, -1 oder -63. Diese Werte lassen sich ganz einfach einer Variablen zuweisen; man kann sie aber auch diese direkt mit "echo" ausgeben.





PHP – Grundsyntax

5.2.2 Seite 12

```
<?php
$zah] = 5;
echo "Die Zah] hat den Wert ".$zah]";
<br />
echo 100;
?>
```

Float-Zahlen sind Zahlen mit Nachkommastellen, wie 345.564, 2.56, 5.0 oder - 34.204235. Wie die Integer-Zahlen können auch die Float-Zahlen in Variablen gespeichert werden und auch mit "echo" ausgegeben werden.

```
<?php

$g = 9.81;

echo "Die Erdbeschleunigung beträgt ".$g." m/s<sup>2</sup>";

?>
```

#### **Rechnen mit Zahlen**

Die Grundrechenarten sind Addition (+), Subtraktion (-), Multiplikation (\*) und Division (/). Der Doppelpunkt (:) ist nicht das Divisionszeichen, denn das ist schon belegt. In PHP und in den meisten anderen Programmiersprachen wird der Schrägstrich (/) als Divisionszeichen benutzt. Ein paar Beispiele:

```
<?php
    echo 4+5; // gibt 9 aus
    echo 4*5; // gibt 20 aus
    echo 4-5; // gibt -1 aus
    echo 4/5; // gibt 0.8 aus
?>
```

Natürlich gibt es in PHP Punkt-vor-Strich-Rechnung. Wenn man eine Zahl durch eine Summe teilen möchte, muss man um die Summe eine Klammer setzen.

```
<?php
    $zah] = 6/3+3;
    echo "Ergebnis: ".$zahl."\n";
    // Ausgabe ist 5
    // 6/3=2 -> 2+3=5
    $zah] = 6/(3+3);
    echo "Ergebnis: ".$zahl."\n";
    // Ausgabe ist 1
    // (3+3) = 6 -> 6/6=1
?>
```





# Zuweisungsoperatoren

PHP kennt eine Menge von Zuweisungsoperatoren, die den Inhalt einer Variable direkt bearbeiten. Dies erhöht die Übersichtlichkeit.

<?php \$var = 5; \$var = \$var + 20; // zu einem Wert 20 addieren \$var += 20; // zu einen Wert 20 addieren ?>

Die obere Variante ist sicherlich schon bekannt. Die untere Variante ist eine Möglichkeit, eine Variable direkt zu verändern. In diesem Fall wird zur Variable "\$var" der Wert "20" addiert. Das Plus (+) vor dem Ist-Gleich (=) gibt an, dass der Wert rechts vom Gleichheitszeichen zur Variable links vom Gleichheitszeichen addiert werden soll. Das Pluszeichen und das Gleichheitszeichen muss zusammengeschrieben werden, damit es als Zuweisungsoperator erkannt wird.

<?php \$var = 5; \$var += 20; // richtig \$var + = 20; // falsch ?>

Dies geht nicht nur mit Plus (+), sondern auch mit allen anderen Grundrechenarten. Beim Dividieren muss man natürlich aufpassen, dass man nicht durch 0 teilt. PHP erzeugt dann einen Fehler.

```
<?php
    $var = 20;
    $var += 4; // $var=24
    $var *= 4; // $var=96
    $var -= 4; // $var=92
    $var /= 4; // $var=23
    $var %= 4; // $var=3 (23/4=5 Rest 3)
?>
```

Die selbe Methode ist auch mit dem Verkettungsoperator möglich. Dabei wird der rechte Ausdruck der Variablen links angehängt.





PHP – Grundsyntax

5.2.2 Seite 14

```
<?php
$x = 5;
$string = "Hallo \n";
$string .= "So kann man gut Sätze anhängen.\n";
$string .= "test";
$string .= "Die Variable hat den Wert ".$x.".\n";
?>
```

Es gibt Fälle, da möchte man den Inhalt einer Variable nur um 1 erhöhen oder verringern. Wie in C, RPG etc. gibt es in PHP dafür eine bestimmte Schreibweise.

```
<?php
$name = 10;
$name++; // erhöhen um 1
echo $name; // gibt 11 aus
?>
```

Syntaxen wie \*\* oder // gibt es nicht. Diese Kurzschreibweisen nennt man Inkrementieren (++) bzw. Dekrementieren (--); sie werden dann benutzt, wenn in einer Schleife gezählt werden muss.

#### 5.2.2.3.1.4 Schleifen

Wir werden in unserem Script alle Datensätze mittels SQL in ein Array einlesen. Dieses Array muss dann Reihe für Reihe in das PHP-Script eingelesen werden.

Dazu benötigen wir eine Schleife. PHP kennt insgesamt vier Schleifenformen, wobei wir uns in diesem Abschnitt auf zwei Schleifenformen konzentrieren:

- Kopfgesteuerte Schleife
- Fußgesteuerte Schleife



#### While-Schleife – kopfgesteuert

Die While-Schleife ist wie folgt aufgebaut:

```
<?php
while (bedingung) {
    // Programmcode
  }
    //bzw.
while (bedingung)
    ein_befehl();
?>
```

Dort wo "bedingung" steht, wird ein Bedingungsausdruck eingetragen, wann beziehungsweise wie lange die While-Schleife ausgeführt werden soll. Wenn die Bedingung wahr (true) ist, wird der Schleifenteil ausgeführt. Wenn die Bedingung falsch (false) ist, dann wird der Schleifenteil logischerweise nicht ausgeführt. Wenn PHP am Ende angekommen ist, springt PHP wieder zu "while" (bedingung) und prüft die Bedingung nachträglich. Wenn diese wieder wahr (true) ist, wird wieder der Schleifenteil ausgeführt. Nach dem Ende des Schleifenteils sieht PHP sich wieder die Bedingung an. Wenn diese "true" ist wird wieder der Schleifenteil ausgeführt. PHP führt also den Schleifenteil solange aus, bis die Bedingung "true" ergibt. "While" (deutsch: während) bedeutet in der Programmiersprache "solange". "Solange die Bedingung wahr (true) ist, führe den Schleifenteil aus". Die Bedingung sollte kein konstanter Ausdruck sein, sondern ein veränderlicher mit Variablen, deren Inhalt sich in der While-Schleife ändert. Denn sonst kann es passieren, dass Sie eine Endlosschleife formulieren, die nicht aufhört, bis der Web-Server das Script nach 30 Sekunden automatisch beendet.

```
<?php

$x = 4;

while (0 < $x) {

echo "Der Wert ist ".$x."<br />\n";

$x--;

}

?>
```



5.2.2 Seite 15



PHP – Grundsyntax

## Do-While-Schleife – fußgesteuert

Seite 16

5.2.2

Die Do-While-Schleife ist wie folgt aufgebaut:

```
<?php
    do {
        // PHP-Code
    }
    while (bedingung);
    // bzw.
    do
        ein_befehl();
    while (bedingung);
?>
```

Das Semikolon hinter "do-while" ist hier notwendig, bei der normalen While-Schleife ist es eher unerwünscht. Der Unterschied der Do-While-Schleife zur While-Schleife ist der, dass die Do-While-Schleife mindestens einmal ausgeführt wird, da die Überprüfung der Bedingung erst am Schleifen-Ende erfolgt. Wenn in der Bedingung "false" steht, wird diese Schleife genau einmal durchgeführt.

```
<?php

$x=2;

do {

echo "Der Wert ist ".$x."<br />\n";

$x--;

}

while (0 < $x);

?>
```

#### 5.2.2.3.2 Benötigte PHP-Funktionen

Neben den reinen PHP-Anweisungen, verfügt PHP auch über einen großen Satz an Funktionen, mit denen der Programmierer wiederum viele spezielle Funktionen ausführen kann. Um unsere Aufgabe zu erfüllen, benötigen wir eine Reihe von Funktionen, mit denen wir über die ODBC-Schnittstelle auf die DB2 des Systems i zugreifen können.



PHP – Grundsyntax



Folgende PHP-Funktionen kommen dabei zum Einsatz:

#### 5.2.2.3.2.1 odbc\_connect()

odbc\_connect --Baut die Verbindung zu einer ODBC-Datenquelle auf.

# Beschreibung

int odbc\_connect (string dsn, string user, string password [, int cursor\_type] )

Liefert eine ODBC-Verbindungskennung *connection\_id* oder 0 (**FALSE**) bei Fehlfunktion zurück.

Die Verbindungskennung *connection\_id* wird von den anderen ODBC-Funktionen benötigt. Es können mehrere ODBC-Verbindungen nebeneinander bestehen. Der optionale vierte Parameter *cursor\_type* setzt die Art des Cursors, der für diese Verbindung genutzt werden soll. Normalerweise wird er nicht benötigt, kann aber bei manchen ODBC-Treiberproblemen nützlich sein.

Folgende Konstanten sind als Cursortyp definiert:

- SQL\_CUR\_USE\_IF\_NEEDED
- SQL\_CUR\_USE\_ODBC
- SQL\_CUR\_USE\_DRIVER
- SQL\_CUR\_DEFAULT

#### 5.2.2.3.2.2 odbc\_do() / odbc\_exec()

odbc\_exec --Bereitet einen SQL-Befehl auf und führt ihn aus.

## Beschreibung

int **odbc\_do** (int connection\_id, string query\_string )

Liefert bei einem Fehler FALSE zurück, sonst ODBC-Ergebniskennung *result\_id*.

odbc\_do() sendet einen SQL-Befehl zum Datenbank-Server, der durch die Verbindungskennung connection\_id bezeichnet wird. Dieser Parameter muss durch odbc\_connect() oder odbc\_pconnect() erzeugt worden sein.



5.2.2 Seite 17



PHP – Grundsyntax

5.2.2 Seite 18

#### 5.2.2.3.2.3 odbc\_fetch\_row()

odbc\_fetch\_row --Liefert eine Datenzeile zurück.

#### Beschreibung

int odbc\_fetch\_row (int result\_id [, int row\_number] )

Wenn **odbc\_fetch\_row()** erfolgreich durchgeführt wurde (also eine oder mehrere Datenzeilen gefunden wurden), dann wird **TRUE** zurückgeliefert. Wenn keine (weitere) Zeile vorhanden ist, wird **FALSE** zurückgegeben.

odbc\_fetch\_row() stellt eine Zeile der von odbc\_do() zurückgegebenen Daten bereit. Nachdem odbc\_fetch\_row() aufgerufen wurde, können die Felder dieser Zeile mit odbc\_result() genutzt werden.

Wenn *row\_number* nicht definiert ist, versucht **odbc\_fetch\_row**() die nächste Zeile aus dem Abfrageergebnis *result\_id* zu holen. **odbc\_fetch\_row**() kann abwechselnd ohne und mit dem Parameter *row\_number* genutzt werden.

Um ein Abfrageergebnis mehrfach zu durchlaufen, kann man **odbc\_fetch\_ row**() mit *row\_number* gleich 1 aufrufen, um dann wie gewohnt mit **odbc\_ fetch\_row**() ohne Zeilenparameter fortzufahren. Wenn der ODBC-Treiber die Herausgabe von bestimmten Zeilen nicht unterstützt, wird der Parameter ignoriert.

#### 5.2.2.3.2.4 odbc\_result()

odbc\_result --Erlaubt den Zugriff auf die Ergebnisdaten.

## Beschreibung

string odbc\_result (int result\_id, mixed field )

Liefert den Inhalt eines bestimmten Feldes zurück. Der Feldparameter *field* kann entweder der Integerwert der Spaltennummer oder ein String mit dem Spaltennamen sein.

Als Beispiel:

\$produkt\_3 = odbc\_result (\$Query\_ID, 3 ); \$produkt\_wert = odbc\_result (\$Query\_ID, "wert");

Ein ITP Handbuch



Seite 19

PHP – Grundsyntax

Der erste Aufruf von odbc\_result() liefert den Wert des drittes Feldes im aktuellen Datensatz im Abfrageergebnis zurück. Der zweite Funktionsaufruf gibt den Wert der Spalte "wert" im aktuellen Datensatz zurück. Wenn die Spaltennummer kleiner als 1 oder größer als die Spaltenzahl ist, tritt ein Fehler auf. Gleiches gilt für einen Spaltennamen, der im Abfrageergebnis nicht vorkommt.

Die Spaltennummern beginnen mit 1.

#### 5.2.2.3.2.5 odbc\_close()

odbc\_close --Beendet eine ODBC-Verbindung.

#### Beschreibung

void odbc\_close (int connection\_id )

**odbc\_close**() beendet die Verbindung zum Datenbank-Server, die durch das Verbindungskennzeichen *ü* gekennzeichnet wird.

#### Anmerkung

Diese Funktion scheitert, wenn auf dieser Datenbankverbindung noch Transaktionen ausstehen. In diesem Fall bleibt die Verbindung bestehen.





PHP – Grundsyntax

#### 5.2.2 Seite 20

#### 5.2.2.3.3 Das dokumentierte Script

Sollte Ihnen der Aufbau einer HTML-Tabelle nicht klar sein, so lesen Sie bitte das Kapitel 5.3 ff.!

<html>

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>ODBC01</title>
```

</head>

```
<body bgcolor="#FFFFF">
<b>Kdnr</b>
<b>Kdnr</b>
<b>Nachname</b>
<b>Strasse</b>
<b>Strasse</b>
<b>Stadt</b>
<b>Stadt</b>
<b>LZ</b>
<b>Kreditgrenze</b>
<b>Kreditcode</b>
<b>Endsumme</b>
<b>Endsumme</b>
<b>Offene Posten</b>
```

<?php

// Variablen deklarieren

\$system = "EPOSI501"; \$user = "LGA01"; \$passwd = "LGA01"; \$pfad = "QIWS.QCUSTCDT";

// Datenbankverbindung herstellen

// odbc\_connect erzeugt bei erfolgreicher Verbindung eine Verbindungs-ID
// in der Variablen \$connectionstring.

// Kann die Verbindung nicht hergestellt werden, so wird false
// zurückgegeben

\$connectionstring = odbc\_connect(\$system, \$user, \$passwd);

// SQL in Befehlsstring stellen

```
// Achtung! Der Dateizugriffspfad wird an den SQL-Befehl
// angekettet
```

PHP – Grundsyntax



Seite 21

\$Query = "SELECT CUSNUM, LSTNAM, STREET, CITY, ZIPCOD, CDTLMT, CHGCOD, BALDUE, CDTDUE FROM ". \$pfad; // Abfrage ausführen // odbc\_do / odbc\_exec führt für die eröffnete Verbindung // die Abfrage aus. // Das Ergebnis der Abfrage wird in Form eines Arrays // (\$queryexe) zurückgegeben \$queryexe = odbc\_do(\$connectionstring, \$Query); // Abfrageergebnis verarbeiten // odbc\_fetch\_row setzt einen Zeiger auf das erste Element // des Array \$queryexe. Bei jedem Durchgang durch die // Schleife wird der Zeiger auf das nachfolgende Element des // Arrays positioniert. // odbc\_fetch\_row gibt false zurück, wenn keine weiteren // Elemente in \$queryexe vorhanden sind. Die Schleife wird // beendet. while(odbc\_fetch\_row(\$queryexe)) { // odbc\_result liest ein Feld aus der aktuellen Zeile des // Array \$queryexe aus und stellt es in eine Zielvariable. // Die einzelnen Felder können über eine Positionsnummer // adressiert werden. \$cusnum = odbc\_result(\$queryexe, 1); \$lstnam = odbc\_result(\$queryexe, 2); \$street = odbc\_result(\$queryexe, 3); = odbc\_result(\$queryexe, 4); \$city \$zipcode = odbc\_result(\$queryexe, 5); \$cdtlimit = odbc\_result(\$queryexe, 6); \$chgcod = odbc\_result(\$queryexe, 7); = odbc\_result(\$queryexe, 8); \$baldue \$cdtdue = odbc\_result(\$queryexe, 9); // Die mit Inhalt gefüllten Variablen werden zeilenweise in

// Die mit Inhalt gefüllten Variablen werden zeilenweise
// eine HTML-Tabelle ausgegeben

Web-Programmierung PHP – Grundsyntax

# 5.2.2

Seite 22

```
print ("");
print ("
                  $cusnum");
print ("
                  $lstnam");
print ("
                  $street");
print ("
                  $city");
print ("
                  $zipcode");
print (" $cdtlimit");
print ("
                  $chgcod");
print (" $baldue");
print (" $cdtdue");
print ("");
```

}

//Datenbankverbindung beenden für die geöffnete Verbindung

odbc\_close(\$connectionstring);

?>

</body> </html>

#### 5.2.2.3.4 Das undokumentierte Script

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>ODBC01</title>
</head>
<body bgcolor="#FFFFFF">
<b>Kdnr</b>
 <b>Nachname</b>
 <b>Strasse</b>
 <b>Stadt</b>
 <b>PLZ</b>
 <b>Kreditgrenze</b>
 <b>Kreditcode</b>
 <b>Endsumme</b>
 <b>Offene Posten</b>
 <?php
```

PHP – Grundsyntax



Seite 23

```
$system = "EPOSI501";
   $user = "LGA01";
   $passwd = "LGA01";
   $pfad = "QIWS.QCUSTCDT";
   $connectionstring = odbc_connect($system, $user, $passwd);
   $Query = "SELECT
                        CUSNUM,
                        LSTNAM,
                        STREET,
                        CITY,
                        ZIPCOD,
                        CDTLMT,
                        CHGCOD,
                        BALDUE,
                        CDTDUE
               FROM ".$pfad;
   $queryexe = odbc_do($connectionstring, $Query);
   while(odbc_fetch_row($queryexe))
   {
        $cusnum = odbc_result($queryexe, 1);
        $lstnam = odbc_result($queryexe, 2);
        $street = odbc_result($queryexe, 3);
               = odbc_result($queryexe, 4);
        $city
        $zipcode = odbc_result($queryexe, 5);
        $cdtlimit = odbc_result($queryexe, 6);
        $chgcod = odbc_result($queryexe, 7);
        $baldue = odbc_result($queryexe, 8);
        $cdtdue = odbc_result($queryexe, 9);
        print ("");
        print ("
                                 $cusnum");
        print ("
                                 $lstnam");
        print ("
                                 $street");
        print ("
                                 $city");
        print ("
                                 $zipcode");
        print (" $cdtlimit");
        print ("
                                 $chgcod");
        print (" $baldue");
        print (" $cdtdue");
        print ("");
   }
   odbc_close($connectionstring);
</body>
</html>
```



?>



PHP – Grundsyntax

# 5.2.2 Seite 24

#### 5.2.2.3.5 Die Ausgabe des Scripts

Folgende Ausgabe wird durch das Script ODBC01.php erzeugt:

🕴 ODBCO1	- Mozilla Firef	DX								
Datei Bea	arbeiten <u>A</u> nsich	nt <u>⊂</u> hronik <u>L</u> esezeid	hen E <u>x</u> tra	as <u>H</u> ilfe						
<					h/odbc01.php			•	Google	Q
🗀 PHP [	🗅 PHP 🔄 HP EPOS 🛅 EPOS Intern 🔄 PhpMyAdmin 🏧 iSeries Access für W 📋 www.midrangemaga 🛅 Javascript 🌻 SELFHTML 8.1.1 (HT 📄 #php/QuakeNet Tut									
ODBC	D1									-
Kdnr	Nachname	Strasse	Stadt	PLZ	Kreditgrenze Kr	reditcode E	ndsumme	Offene Posten		
938472	Henning	4859 Elm Ave	Dallas	75217	5000 3		37.00	0.00		
839283	Jones	21B NW 135 St	Clay	13041	400 1		100.00	0.00		
392859	Vine	PO Box 79	Broton	5046	700 1		439.00	0.00		
938485	Johnson	3 Alpine Way	Helen	30545	9999 2		3987.50	33.50		
397267	Tyron	13 Myrtle Dr	Hector	14841	1000 1		0.00	0.00		
389572	Stevens	208 Snow Pass	Denver	80226	400 1		58.75	1.50		
846283	Alison	787 Lake Dr	Isle	56342	5000 3		10.00	0.00		
475938	Doe	59 Archer Rd	Sutter	95685	700 2		250.00	100.00		
693829	Thomas	3 Dove Circle	Casper	82609	9999 2		0.00	0.00		
593029	Williams	485 SE 2 Ave	Dallas	75218	200 1		25.00	0.00		
192837	Lee	5963 Oak St	Hector	14841	700 2		489.50	0.50		
583990	Abraham	392 Mill St	Isle	56342	9999 3		500.00	0.00		

ODBC01.php – Ausgabe





# 5.2.3 Seite 1

# 5.2.3 PHP und i5/OS-Befehle

#### 5.2.3.1 runQry mit PHP? runQry mit PHP!

#### 5.2.3.1.1 Vorbereitende Gedanken

Auf vielen Systemen, die ich auf meinen Wegen durch die verschiedensten Unternehmenslandschaften kennengelernt habe, findet sich eine Fülle von Abfragen, die sich im Zuge der Zeit angehäuft haben. Das bedeutet selbstverständlich nicht (zwingend), dass es an der Zeit ist, diese Bestände zu durchforsten und vom Unterholz zu befreien, sondern eher, wie die Reichhaltigkeit der Auswertungsvarianten und Möglichkeiten auch in der näheren Zukunft weiter genutzt werden kann. Und zwar auch eingebettet in eine mit Webtechnologie durchsetzte IT der Moderne.

Ein sicher elementares Szenario ist ein Query, dessen Ergebnis in eine Datei in einer bestimmten Bibliothek geschrieben wird. Die Sätze dieser Datei werden dann in unterschiedlicher Art und Weise weiter verwendet, sei es als Input für weitere Queries, für einen Listendruck, wie auch immer.

#### 5.2.3.1.2 Empfohlenes Vorwissen

Zum besseren Verständnis dieses Kapitels sollten Sie schon ein grundsätzliches Interesse für PHP besitzen und natürlich mit dem Umgang mit Befehlen auf der iSeries vertraut sein. In den Abschnitten, in denen es um die Restrukturierung des Codes geht, wird ein wenig Kenntnis der UML (Unified Modelling Language) hilfreich sein. Ich werde dort ansetzen, den Code in eine objektorientierte Form zu bringen; für ein tiefergehendes Verständnis sind Vorkenntnisse der Objektorientierung notwendig, da ich die Methodik dieser Art der Softwareentwicklung nicht eingehend besprechen werde.

PHP und i5/OS-Befehle

5.2.3 Seite 2

**Cabite** 

#### 5.2.3.1.3 Aufruf einer vorhandenen Abfrage

Abfrage ausführen	(RUNQRY)
Auswahl eingeben und Eingabetaste drücken.	
Abfrage	Name, *NONE Name, *LIBL, *CURLIB
Datei	Name, *SAME Name, *RUNOPT, *LIBL, *CURLIB Name, *RUNOPT, *FIRST
+ für weitere Werte	*RUNOPT, *DISPLAY *RUNOPT, *DETAIL, *SUMMARY *NO, *YES

Diese Abfrage liefert, im kleinen Ausschnitt, folgendes Ergebnis.

				Bericht anzeigen		
Abfrag		BOEHMER/I	DRY001		Breite des B	Berichts .: 656
Anfang auf Zeile					Anfang in Spalte	
Zeile	+.	1+	2+3+	.4+5+6+7	+8+9+10	+11+12
	Firma	Identnr.	Anrede Kennzeichen	Name 1	Name 2	Name 3
000001		1.000.200		BT Deutschland GmbH		
000002		1.000.500		Coty Deutschland GmbH		Fort Malakoff Par
000003		1.001.200		ADP Employer Services GmbH		
000004		1.001.500		Händle GmbH & Co. KG	Maschinen und Anlagenbau	
000005		1.001.700		F.W. Hempel & CO.	Erze und Metalle	
000006		1.001.900		Unilog Integrata Training AG	Computer und Management	Trainings GmbH

Soll die Abfrage aus einem PHP-Script heraus aufgerufen werden, so darf sie keine interaktive Ausgabe vorsehen, sondern die Ergebnisdaten sollen in eine Ausgabedatei geschrieben werden.

Abfrage ausführen (RUNQRY)						
Auswahl eingeben und Eingabetaste drücken.						
Abfrage	. > <u>QRY001</u>	Name, *NONE				
Bibliothek	. > <u>BOEHMER</u>	Name, *LIBL, *CURLIB				
Abfragedatei:						
Datei sa kana kana kana	i	Name, *SAME				
Bibliothek	. <u>*LIBL</u>	Name, *RUNOPT, *LIBL, *CURLIB				
Teildatei	*FIRST	Name, *RUNOPT, *FIRST				
+ für weitere Werte						
Berichtsausgabeart	. > <u>*OUTFILE</u>	*RUNOPT, *DISPLAY				
Ausgabeform	. > <u>*DETAIL</u>	*RUNOPT, *DETAIL, *SUMMARY				
Satzauswahl	. <u>*NO</u>	*NO, *YES				
Abfragedefinition drucken	*RUNOPT	*RUNOPT, *NO, *YES				
Ausgabedatei:						
Datei	. > FLEQRY001	Name, *RUNOPT				
Bibliothek	. > <u>BOEHMER</u>	Name, *RUNOPT, *CURLIB				
Teildatei	. > <u>*FIRST</u>	Name, *RUNOPT, *FIRST				
Auswahl	. > *NEWFILE	*RUNOPT, *NEWFILE				



PHP und i5/OS-Befehle



Seite 3

Der erste Teilschritt wird der entsprechend parametrisierte Aufruf des Befehls **RUNQRY** sein. Also ist festzustellen, wie ein Befehl prinzipiell mit PHP aufrufbar ist:

i5\_command bool i5\_command(string command[, array inputs, array outputs, resource connection]). Calls CL command.

```
👜 script001.php 🗙
   <?php
   // herstellen einer Verbindung.
   $server = '172.16.110.22';
   $user = 'BOEHMER';
   $password = 'RONALD';
   $Connection = i5 connect( $server, $user, $password);
   // einfacher html-Rumpf
   echo '<html>':
   // prüfen, ob Verbindung erstellt werden konnte oder nicht.
   // Passender Kommentar wird ausgegeben und eine geöffnete
   // Verbindung wieder geschlossen.
   if( !$Connection ){
       echo 'sorry, connection failed!';
   }
   else{
      echo 'connected, congratulations!';
       i5_close( $Connection );
   }
   echo '</html>';
   ?>
```

Die obige Abbildung stellt ein Stück trivialen Codes dar, der noch einmal die Herstellung einer Verbindung zu einer iSeries verdeutlicht.

Ich werde nun die Quelle erweitern, so dass der RUNQRY parametrisiert aufgerufen werden kann. Sollte bei der Ausführung etwas fehlschlagen, soll ein Fehlertext ausgegeben werden.

Der Aufruf soll folgende Form annehmen.

```
$done = i5_command( $Command, $arrInputs, $arrOutputs, $Connection );
if( !$done ) echo i5_errormsg( $Connection );
i5_close( $Connection );
```

Die erste Zeile führt den eigentlichen Befehl auf der iSeries aus. I5\_command gibt einen booleschen Wert zurück, im Fehlerfall FALSE, wenn der Befehl korrekt abgelaufen ist, TRUE. Also kann über den Inhalt von \$done festgestellt werden, ob eine Meldung vorliegt oder nicht.



PHP und i5/OS-Befehle

# 5.2.3 Seite 4

Bei einem Fehlschlag, also \$done == false, gebe ich über i5\_ errormsg(\$Connection) die zugrundeliegende Meldung aus. Nach dem kompletten Ablauf des Scripts werden die Verbindungsressourcen wieder freigegeben. Schauen wir uns die einzelnen Parameter des Aufrufs an.

In \$Command wird der Name des Befehls übergeben. Das Array \$arrInputs nimmt die benötigten Aufrufparameter auf, \$arrOutputs (ebenfalls ein Array) erhält die Rückgabeparameter (bei diesem Befehl sind keine vorhanden). Zuletzt wird in \$Connection noch die notwendige Verbindungsressource zum Zielsystem definiert.

Stellen wir zunächst fest, wie der Befehl sich aufbaut, wenn er über die Befehlszeile wie gewohnt eingegeben wird.

```
Auswahl oder Befehl
===> <u>RUNQRY QRY(BOEHMER/QRY001) OUTTYPE(*OUTFILE) OUTFORM(*DETAIL) OUTFILE(BOEH</u>
MER/FLEQRY001)
```

Es wird ein assoziatives Array aufgebaut, dessen Elementschlüssel den Namen der zu belegenden Parameter darstellen und dessen zugeordnete Elementwerte die den Befehlsparametern eingestellten Werte beinhalten. Welche Elementschlüssel und welche assoziierten Inhalte werden benötigt?

Aufbau und Inhalt von \$arrInputs

Schlüssel	Wert
QRY	BOEHMER/QRY001
OUTTYPE	*OUTFILE
OUTFORM	*DETAIL
OUTFILE	BOEHMER/FLEQRY001

Die Commandvariable wird mit 'RUNQRY' belegt. In der folgenden Abbildung sehen Sie die Umsetzung in Code.





PHP und i5/OS-Befehle



Seite 5

Um das Script durchzuführen, öffne ich einen Webbrowser und gebe die URL für den Seitenstart ein.

D.	http://172.16.110.22/basics/script001_1.php
*	<pre>/// // // // // // // // // // // // //</pre>

Der Anzeige nach scheint das Script erfolgreich abgelaufen zu sein. Demzufolge sollte in der Bibliothek BOEHMER jetzt die physische Datei FLEQRY001 zu finden sein. Schauen wir nach, ob die Datei erstellt wurde und welchen Inhalt sie hat.

Mit Objekten arbeiten						
Auswahl eingeben und Eingabetaste drücken.						
2=Berechtigung editieren 3=Kopieren 4=Löschen 5=Berechtigung anzeigen						
7=Umbenennen 8=Beschreibung anzeigen 13=Beschreibung ändern						
Objekt	Art	Bibliothek	Attribut	Text		
FLEQRY001	*FILE	BOEHMER	PF			
	nl eingeben erechtigung mbenennen Objekt FLEQRY001	nl eingeben und Eingab erechtigung editieren mbenennen 8=Beschrei Objekt Art FLEQRY001 *FILE	Mit Objekto nl eingeben und Eingabetaste drück erechtigung editieren 3=Kopieren nbenennen 8=Beschreibung anzeign Objekt Art Bibliothek FLEQRY001 *FILE BOEHMER	Mit Objekten arbeiten al eingeben und Eingabetaste drücken. erechtigung editieren 3=Kopieren 4=Löscher abenennen 8=Beschreibung anzeigen 13=Besch Objekt Art Bibliothek Attribut FLEQRY001 *FILE BOEHMER PF		

Datei-Objekt

Physische Teildat Datei : FLEQRYOO1 Teildatei : FLEQRYOO1 Steuerung : Suchen :	ei anzeigen Bibliothek : BOEHMER Satz : 1 Spalte : 1	
<u>*+</u> 1+2+3+	.4+5+6+	7+8+9+0
BT Deutschland GmbH Coty Deutschland GmbH ADP Employer Services GmbH		Fort Malakoff Park
& Händle GmbH & Co. KG ø F.W. Hempel & CO. ° Unilog Integrata Training AG Lloyd Shoes GmbH & Co. KG	Maschinen und Anlagenbau Erze und Metalle Computer und Management	Trainings GmbH

Inhalt der Datei (Ausschnitt)

Die Dinge sehen gut aus, auch die Überprüfung der Anzahl der bereitgestellten Sätze fällt positiv aus.



Ergänzung 1/2007

PHP und i5/OS-Befehle

# 5.2.3 Seite 6

#### 5.2.3.1.4 Datensätze auslesen (native)

In einem neuen Script sollen die Datensätze der Ausgabedatei abgerufen und in einer HTML-Tabelle auf einer Webseite angezeigt werden. Der Zugriff auf die Daten der iSeries soll über einen nativen Dateizugriff erfolgen, dazu stelle ich Ihnen vorab die benötigten Befehle vor.

Die Datei muss geöffnet werden, bevor auf sie zugegriffen wird. Hier werde ich mit reinen lesenden Operationen auskommen und gebe den speziellen Dateimodus I5\_OPEN\_READ an, der aber auch den Defaultmode darstellt. Der Befehl lautet i5\_open und wird mit der folgenden Syntax aufgebaut:

Nat	ive File Acces	55	
i5_o	pen		
res	ource i5_ope	en (string fileName [, int mode ][,resource connection	on]).
Open	s native i5 file.		
Retu	rns: Resource, if "	open" is successful, false otherwise.	
Argu	ments:		
	name	File name, may include library	
	mode	File mode to use:	
I5_OPEN_READ - <u>default</u>		I5_OPEN_READ - <u>default</u>	
		I5_OPEN_READWRITE	
		I5_OPEN_COMMIT	
		I5_OPEN_SHRRD	
		I5_OPEN_SHRUPD	
		I5_OPEN_SHRNUPD	
		I5_OPEN_EXCLRD	
		I5_OPEN_EXCL	
	connection	Connection - result of i5_connect	



Wenn die Dateiressource nicht benötigt wird, kann sie über den Befehl i5\_free\_file freigegeben werden. Auch dazu sehen Sie das Syntaxdiagramm.

15_T	ree_file	2			
boo	ol i5	free file (	resource	file	).
Clos	es file ha	- undle and frees file	resources		
		indic and nees me	resources.		
Retu	arns: Bo	olean success valu	e.		
	trans anote t				
Argi	imenis:				
Argı	imenis:	20000000			
Argı.	file	i5 file resource			

Damit Sie diese Funktionen einmal im Zusammenhang sehen, habe ich ein Script aufgebaut, das die Queryausgabedatei öffnet und wieder schließt und natürlich auch vorher eine Verbindung zum System herstellt und diese ebenfalls nach Beendigung des Scripts wieder schließt. Bei den einzelnen Schritten wird jeweils die Ausgabe einer Fehlermeldung vorgesehen. Damit habe ich einen prinzipiellen Rumpf bereitgestellt, der in den nächsten Schritten weiter ausgebaut wird. Hier nun das Script.

```
if ( $Connection ) {
    echo 'Verbindung hergestellt, Daten werden abgerufen.';
    $fileName = 'BOEHMER/FLEQRY001';
    $rsQuery = i5_open( $fileName, I5_OPEN_READ, $Connection );
    if( $rsQuery ) {
        echo 'Datei geöffnet';
        if( i5 free file ( $rsQuery ) ){
            echo 'Datei geschlossen.';
        3
        else{
            echo 'Fehler beim Schließen.';
            echo i5 errormsg( $Connection );
        }
    }
    else{
        echo 'Fehler beim Öffnen';
        echo i5 errormsg( $Connection );
    }
    i5 close( $Connection );
3
else{
    echo 'sorry, connection failed!';
}
```

Script002\_0.php



Ergänzung 1/2007



PHP und i5/OS-Befehle

# 5.2.3 Seite 8

Der Rumpf wird jetzt um eine Schleife erweitert, die über alle Sätze der Datei iteriert. Um die Funktionsfähigkeit zu testen, wird die vierte Spalte jedes einzelnen Satzes mit einem folgenden <br/> br /> ausgegeben. Wenn in der Webseite dann eine Liste von Firmen auftaucht, kann die HTML-Formatierung vorgenommen werden.

Das Beschaffen eines Satzes übernimmt der Befehl *i5\_fetch\_array(\$rsQuery, I5\_READ\_NEXT)*. \$rsQuery ist die Ressource, die von i5\_open... geliefert wurde und I5\_READ\_NEXT eine Konstante, die den nächsten Satz liest.

Die Schleife selbst wird beendet, wenn i5\_fetch... FALSE liefert, in diesem Fall konnte kein weiterer Satz mehr gelesen werden.

Die nächste Abbildung zeigt die Syntax des Befehls i5\_fetch, den es auch in anderen Varianten gibt.

i5_fetch_array		
array i5_f array i5_f object i5_ array i5_f	<pre>etch_array(resource result [, int of etch_assoc(resource result [, int of fetch_object(resource result [, int etch_row(resource result [, int opt</pre>	<pre>option ] ). option ] ). c option ] ). cion] ).</pre>
Fetches a row of d	ata from the resource.	access on abject containing the data
array -	by index and name.	array of an object containing the data.
assoc -	· by name.	
<b>row</b> - b	oy index.	
object	<ul> <li>by name as object properties.</li> </ul>	
Arguments:		
result	Resource resulting from operation returning data	
option	Flag specifying which record to fetch. I5_READ_SEEK Current record I5_READ_NEXT Next record I5_READ_PREV Previous record I5_READ_FIRST First record I5_READ_LAST Last record Default is I5_READ_NEXT	

Hier nun das um die Lese-Schleife erweiterte Script und daran anschließend die Ansicht eines Ausschnitts der Webseite.



PHP und i5/OS-Befehle







Erweitertes Script

Contemporal States (%) 🗸 🖉 🖉	cript002_1.php
r 🛠 🔡 🗸 🌈 http://172.16.110.22/ba	. × Gi5_connect - Google-Suche
/erbindung hergestellt, Daten werden abgeru oty Deutschland GmbH ADP Employer Services GmbH	fen.Datei geöffnetBT Deutschland GmbH
ländle GmbH & Co. KG .W. Hempel & CO.	eine Spalte des Satzes

Ausgabe Webseite

PHP und i5/OS-Befehle

# 5.2.3 Seite 10

#### 5.2.3.1.5 HTML-Tabelle aus Datensätzen generieren

Nun sollen die Daten noch dynamisch in eine HTML-Tabelle geschrieben werden. Da im bestehenden Script nur das <html>...</html> – Tag ausgegeben wurde, schreibe ich noch <body> mit in die Ausgabe.

Die oben beschriebene Schleife über alle Sätze wird von dem Tabellen-Tag ... umgeben und jeder einzelne Satz wiederum von ...umgeschlossen, um die HTML-Zeile zu definieren.

Zusätzlich muss die Anzahl der Spalten ermittelt werden, da auch für sie eine Schleife benötigt wird, um den Spalteninhalt mit umschließenden ...

```
// Schleife über alle Sätze
$arrDataRow = i5_fetch_array( $rsQuery, I5_READ_NEXT );
$colCount = count( $arrDataRow );
echo '';
while ( $arrDataRow ) {
    echo '';
    for( $i = 0; $i < $colCount; $i++ ){
        echo '<td>';
    }
    echo '';
}
echo '';
}
echo '';
$arrDataRow = i5_fetch_array( $rsQuery, I5_READ_NEXT );
}
echo '';
echo '';
echo '';
```

HTML-Tabelle generieren

#### Achtung!

Da die Daten ohne Berücksichtigung der Gesamtzahl der in der Datei vorhandenen Sätze gelesen werden, ist bei diesem einfachen Aufbau unter Umständen mit Performance-Einbußen zu rechnen. Überlegungen, die sich auf ein blockweises Beschaffen mit entsprechenden Nachlese-Aktionen beziehen, sollten nicht allzu weit beiseite gelegt werden!

Es fehlen noch die Spaltenüberschriften, damit die Tabelle etwas ansehnlicher wird. Auch diese Informationen können aus den Metadaten der Datensätze ermittelt werden. Für diesen Zweck kann der Befehl *i5\_field\_name* verwendet werden.



Seite 11



In meinem Script werden die Spaltenheader ausgelesen, das funktioniert ähnlich wie beim Erstellen der Datenzeilen der Tabelle. Eine Schleife iteriert über alle Spalten des Satzes und gibt die Überschrift über *i5\_field\_name* in Spaltenheadertags ...

```
// Spaltenüberschriften herstellen
echo '';
for( $i = 0; $i < $colCount; $i++ ){
    echo '<th>'.i5_field_name( $rsQuery, $i ).'';
}
echo '';
```

#### Spaltenheader erstellen

Wie die so veränderte Ausgabe im Webbrowser aussieht, können Sie der nächsten Abbildung entnehmen.



Ausgabe mit Spaltenüberschriften





PHP und i5/OS-Befehle

5.2.3 Seite 12

Es ist natürlich noch anzustreben, etwas aussagefähigere Bezeichnungen zu verwenden.

Ich möchte jedoch den Code der bisher gezeigten Scripte einem Reengineering unterziehen, damit die vorhandenen Fähigkeiten in den folgenden und etwas komplexeren Szenarien einfacher einsetzbar sind. Dabei gilt mein erster Blick auch dem ersten Script, das die Abfrage auf der iSeries aufrief und die Ausgabedaten in Form der Datei erstellte.

#### 5.2.3.1.6 Restrukturierung und die ersten Klassen

Wie kann ein Command-Aufruf beschrieben werden?

- Es soll ein iSeries-Befehl auf einem Zielsystem aufgerufen werden.
- Dieser Command hat verschiedene Parameter.
- Ein Parameter selbst hat eine Bezeichnung, eine Standardbelegung, eine aktuelle Belegung und optional eine Liste von möglichen vordefinierten Werten.

Es fallen mindestens zwei Kandidaten für Klassen ins Auge, die ich I5Command und I5CommandParameter nennen werde.



Beide Klassen sind in einer separaten PHP-Datei abgelegt und ich werde Ihnen als Erstes den Aufbau der Parameterklasse verdeutlichen. Die Abbildung stellt eine Übersicht der Klasse dar.




5.2.3 Seite 13

```
class I5CommandParameter {
    private $m_Name = null;
    private $m_DftValue = null;
    private $m_ActValue = null;
    private $m_DftValList = null;
    * Konstruktor.[]
    public function __construct( $name, $value ) {[]
    * Gibt den Namen des Parameters zurück.[]
    public function getName() {[]
    * Gibt den aktuellen Wert des Parameters zurück.[]
    public function getValue() {[]
}
```

Die sämtlich als *private* deklarierten Attribute nehmen den Namen des Parameters auf, seinen Standardwert, den aktuellen Wert und in m\_DftValList eine optionale Liste von vordefinierten Werten. Diese Liste kann mit den Vorschlägen eines Prompts über die Funktionstaste F4 in den Befehlsbildschirmen der iSeries verglichen werden.

Über den Konstruktor wird ein Parameter erstellt, der durch \$name und \$value sofort eine Bezeichnung und eine Wertbelegung erhält.

```
* Konstruktor.[]
public function __construct( $name, $value ){
    $this->m_Name = $name;
    $this->m_ActValue = $value;
}
```

Die beiden öffentlichen Methoden sollen nachher im I5Command-Objekt zur Erstellung des Inputparameter-Arrays für den i5\_command(...)-Befehl verwendet werden, der bisher eingesetzt wurde.

Auch dazu zeige ich Ihnen die Implementierung.

```
* Gibt den Namen des Parameters zurück.[]
public function getName() {
    return $this->m_Name;
}
* Gibt den aktuellen Wert des Parameters zurück.[]
public function getValue() {
    return $this->m_ActValue;
}
```

Getter-Implementierung





Web-Programmierung

PHP und i5/OS-Befehle

Betrachten wir jetzt die I5Command-Klasse in der Übersicht.

Seite 14

class I5Command {
 private \$m\_ArrParameter = array();
 private \$m\_Name = null;
 \* Konstruktor...
 public function \_\_construct( \$name ) {...
 \* Fügt einen neuen Parameter hinzu...
 public function addParm( I5CommandParameter \$parm ) {...
 \* Führt den Befehl aus...
 public function run( \$cnc ) {...
 \* Erstellt das Array der Input-Parameter...
 protected function getArrInputs() {...
}

Das Attribut  $m\_ArrParameter$  wird die einzelnen Parameter vom Typ I5CommandParameter aufnehmen, in  $m\_Name$  wird die Bezeichnung des Befehls geschrieben. Hier ist der Name gemeint, der in  $i5\_command(...)$  anzugeben ist. Der Konstruktor erledigt noch nicht mehr, als diesen Namen in das Attribut zu schreiben und wird aus diesem Grund auch nicht abgebildet.

Die Methode *addParm* fügt der Befehlsklasse neue Parameter hinzu und erwartet in der Übergabe Instanzen von I5CommandParameter. Der neue Parameter wird in die Liste (Array) der Befehlsparameter aufgenommen.



#### Methode für neue Parameter

Fassen wir nun die getArrInputs() – Methode ins Auge. Sie dient dazu, das Inputparameter-Array für den *i5\_command*-Befehl zurückzugeben. Da diese Methode nur klassenintern verwendet werden soll, ist sie als geschützt deklariert.





Seite 15

Die benötigten Inputparameter werden als Elemente in das lokale Array *arrInputs* geschrieben.

Eine for-Schleife läuft über alle Elemente der Parameterliste der Klasse, wobei die Anzahl aus der PHP-Funktion *count(...)* ermittelt wird. Wie Sie aus den ersten Scripten noch wissen, liegt ein Parameter für *i5\_command* in der Form array[ parmName ] = parmWert vor. Der parmName wird aus dem i-ten Element der Parameterliste (das ist eine Instanz von I5CommandParameter!) durch die Methode *getName()* ermittelt. Über *getValue()* wird der aktuelle Wert dieser Parameterinstanz gelesen. Damit ist ein Element des Inputparameter-Arrays gefüllt.

Sobald die Schleife alle Parameter der Liste verarbeitet hat, wird das erstellte assoziative Array via *return* an den Aufrufer zurückgegeben.

Der eigentliche Aufruf des Befehls implementiert sich dann sehr einfach, auch wenn die Vorarbeit etwas komplexer war. Nur muss man sich ab sofort nicht mehr um diese Dinge kümmern, sie sind, wie es sich gehört, ab sofort gekapselt.

```
* Führt den Befehl aus...
public function run( $cnc )(
        $arrInputs = $this->getArrInputs();
        $arrOutputs = array();
        $done = i5_command( $this->m_Name, $arrInputs, $arrOutputs, $cnc );
        return $done;
}
```

# Ausführen des Befehls

Die Methode erhält eine Verbindung \$cnc als Parameter, erstellt über die eben besprochene Methode die Eingabeparameter, besetzt ein leeres Ausgabeparameterarray und ruft dann einfach *i5\_command* auf. Der boolesche Rückgabewert wird von run(...) selbst wieder zurückgegeben.



Ergänzung 1/2007



5.2.3 Seite 16 Damit sind wir nun soweit und können das ursprüngliche Script zur Ausführung der Abfrage abändern.

In dieses Script muss noch ein *require\_once(Pfad zur Klassendatei.php)* aufgenommen werden, wenn die beiden neuen Klassen wie bei mir in einer separaten PHP-Quelle abgelegt werden sollen.

```
$RunOry = new ISCommand( 'RUNORY' );
$RunOry->addParm( new ISCommandParameter( 'QRY', 'BOEHMER/QRY001' ) );
$RunQry->addParm( new ISCommandParameter( 'OUTTYPE', '*OUTFILE' ) );
$RunQry->addParm( new ISCommandParameter( 'OUTFORM', '*DETAIL' ) );
$RunQry->addParm( new ISCommandParameter( 'OUTFILE', 'BOEHMER/FLEQRY001 *FIRST *NEWFILE' ) );
$done = $RunQry->run( $Connection );
if( !$done ) echo i5_errormsg( $Connection );
i5_close( $Connection );
```



# 5.2.3.1.7 Ein Webfrontend für den Query-Aufruf

Aufruf des Query, die Bereitstellung der Daten in einer Ausgabedatei und die Darstellung in einer HTML-Tabelle sind implementiert worden. Was noch wünschenswert wäre, ist der Zugriff auf diese Funktionalitäten über eine Weboberfläche. Um den Aufwand ein wenig zu reduzieren, beschränke ich mich bei der Parametrisierung auf die Parameter, die ich bisher variabel gehalten habe. Das betrifft also QRY, OUTFILE und OUTFORM. OUTTYPE lasse ich noch außen vor, weil ich die Dateiausgabe beibehalten möchte. Betrachten wir den 5250-Schirm, wenn der Befehl *RUNQRY* auf der iSeries mit Bedienerführung aufgerufen wird.

Abtrage austuhren (RU	INQRY J
Auswahl eingeben und Eingabetaste drücken.	
Abfrage	Name, *NONE
Bibliothek BOEHMER	Name, *LIBL, *CURLIB
Abfragedatei:	
Datei s construction	Name, *SAME
Bibliothek <u>*LIBL</u>	Name, *RUNOPT, *LIBL, *CURLIB
Teildatei <u>*FIRST</u>	Name, *RUNOPT, *FIRST
+ für weitere Werte	
Berichtsausgabeart > <u>*OUTFILE</u>	*RUNOPT, *DISPLAY
Ausgabeform	*RUNOPT, *DETAIL, *SUMMARY
Satzauswahl <u>*NO</u>	*NO, *YES
Abfragedefinition drucken <u>*RUNOPT</u>	*RUNOPT, *NO, *YES
Ausgabedatei:	
Datei	Name, *RUNOPT
Bibliothek BOEHMER	Name, *RUNOPT, *CURLIB
Teildatei	Name, *RUNOPT, *FIRST
Auswahl <u>*RUNOPT</u>	*RUNOPT, *NEWFILE
	Weitere
F3=Verlassen F4=Bedienerf. F5=Aktualisieren	F12=Abbrechen
F13=Verwendung der Anzeige F24=Weitere Tast	en

Befehlsprompt

Der über eine Webschnittstelle zu steuernde Anteil befindet sich in den kleinen umrahmten Bereichen der Abbildung.

Ich beginne mit dem Entwurf einer Seite, die die geforderten Parameter darstellen kann. Auch hier lege ich keinen Wert auf Design und Eleganz, da lediglich der Command-Unterbau angesteuert werden soll.





Die gerahmten Abschnitte des dargestellten HTML-Codes enthalten Input-Elemente in einem Formular, die bei der gewählten POST-Methode an das für PHP global verfügbare \$\_POST-Array übergeben werden.

Dieses Array kann in der Zielseite, aufgerufen durch das *action*-Attribut des <form> – Tags, von PHP ausgewertet werden und dann die benötigten Zeichenfolgen für den Aufruf des iSeries-Befehls zusammensetzen.

Die nun folgende Abbildung zeigt Ihnen die Darstellung des obigen Codes im Internetexplorer. Die verschiedenen Eingabefelder sind mit Werten ausgefüllt worden.



Web-Programmierung	
PHP und i5/OS-Befehle	



51213 Seite 19

🖉 Abfrage ausführen (	RUNQRY ) - Windows Internet Explorer
🔆 🔁 🗸 🖉 🕹 🔄	2.16.110.22/basics/script003_0.php
🚖 🏟 🌈 Abfrage ausfi	ühren ( RUNQRY )
Abfrage aus Abfrage QRY001 Bibliothek Ausgabeform *DETAIL	führen ( RUNQRY )
Ausgabedatei	
Datei	FLEQRY001
Bibliothek	BOEHMER
ausführen	

Wenn ich jetzt auf "*ausführen*…" klicke, werden die eingetragenen Werte in das \$\_POST-Array eingesetzt und die Seite *script001\_3.php* aufgerufen. Dort ergibt sich folgendes Bild.

🤇 http://	172.16.	110.22/b	asics/script001_3.	php - Windows Inte	rnet Explorer	
90	👻 🙋 hi	ttp://172.16	5.110.22/basics/script00	1_3.php		
* *	🏉 http:/	/172.16.11	0.22/basics/script001_3	.php		
KLFINR	KLIDNR	KLANKZ	KLNAM1	KLNAM2	KLNAM3	KLSTR
1	1001200	1	ADP Employer Services GmbH			Im Hollergrund 3
1	1004300	1	Rotes Kreuz Krankenhaus DRK			StPauli-Deich 24
1	1004400	1	Lloyd Dynamowerke GmbH			Hastedter Osterdeic 250
1	1006900	1	SWING Software GmbH			Bremer Str. 59
1	1007200	1	Wächter & Wächter		Werbeagentur GmbH	Außer der Schleifmühle 65
1	1009000	1	KARL KÖNECKE GmbH & Co. KG	Fleischwarenfabrik		Sebaldsbrücker Bahnhof 1
1	1009800	1	Bindan GmbH & Co.			Bahnhofstr. 8
1	1010100	1	Carl Ungewitter			Bürgermeister-Smidt Str. 56

In welcher Art und Weise die Scripte bearbeitet werden müssen, um dieses Verhalten zu erreichen, ist auch in diesem Fall recht einfach. Ausgehend vom letzten Script, das die neue I5Command und I5CommandParameter-Klassen einsetzt, muss das \$\_POST-Array ausgelesen werden und die Informationen für die Aufrufparameter des *i5\_command*-Befehls zusammengesetzt werden.



Ergänzung 1/2007



```
Web-Programmierung
```

5.2.3 Seite 20 Diesen Code zeige ich Ihnen in der unten stehenden Abbildung.

```
$QRY = $_POST[ 'QRYLIB' ].'/'.$_POST[ 'QRYNAME' ];
$OUTFORM = $_POST[ 'OUTFORM' ];
$OUTFILE = $_POST[ 'FLELIB' ].'/'.$_POST[ 'FLENAME' ].' *FIRST *NEWFILE';
```

In die Variable \$QRY wird der Inhalt von QRYLIB und QRYNAME mit einem Slash verbunden geschrieben. Der Parameter OUTFORM wird einfach übernommen und in der letzten Zeile der Parameter \$OUTFILE zusammengesetzt.

Damit stehen die notwendigen Aufrufinformationen zur Verfügung. Nun muss noch die I5Command-Klasse instanziiert werden und die Parameter mit den oben erstellten Variablen beschickt werden.



Parameter mit Variablen beschicken

Diesen nur leicht abgeänderten Zeilen wird der Code, der schon bisher zur Ausgabe der HTML-Tabelle diente, angeschlossen.





```
5.2.3
$fileName = $OUTFILE;
                                                                          Seite 21
$rsQuery = i5_open( $fileName, I5_OPEN_READ, $Connection );
if( $rsQuery ){
   // Schleife Ober alle SOtze
   $arrDataRow = i5 fetch array( $rsQuery, I5 READ NEXT );
   $colCount = count( $arrDataRow );
   echo '<body>';
   echo '';
   // SpaltenOberschriften herstellen
   echo '';
   for( $i = 0; $i < $colCount; $i++ ){</pre>
      echo ''.i5 field name( $rsQuery, $i ).'';
   }
   echo '';
   // DatensOtze als Html-Tabelle ausgeben
   while ( $arrDataRow ) {
      echo '';
      for( $i = 0; $i < $colCount; $i++ ){</pre>
          echo ''.$arrDataRow[ $i ].'';
      1
      echo '';
      $arrDataRow = i5 fetch array( $rsQuery, I5 READ NEXT );
   }
   echo '';
   echo '</body>';
```



# Seite 22

#### 5.2.3.1.8 Liste aller Queries in einer Bibliothek

Das Webfrontend soll erweitert werden, so dass die Eingabefelder für Bibliothek und auszuführende Abfrage nicht mehr von Hand ausgefüllt werden müssen, sondern eine Liste in einer ComboBox angezeigt wird.

Dafür wird zuerst eine Liste aller Abfragen in einer Bibliothek bereitgestellt. Die Informationen werden in die Datei FLEQUERIES in der Bibliothek BOEHMER gestellt. Dazu habe ich den folgenden Befehl aufgerufen.

```
DSPOBJD OBJ (BOEHMER/*ALL) OBJTYPE (*QRYDFN) OUTPUT (*OUTFILE) OUTFILE (BOEHME
R/FLEQUERIES)
```

#### Liste der Abfragen bereitstellen

Den Inhalt der erzeugten Datei stelle ich in der nächsten Abbildung dar. Aus dieser Datei werden nur die Einträge für "Bibliothek" und "Name der Abfrage" benötigt. Die Namen entnehme ich der Feldbeschreibung über DSPFFD.

					Bericht a	anzeigen	
Anfang Zeile	auf Zeile . +1			+4	+5+.	6+	+
	Anzeige- jahrhundert	Anzeige- datum	Anzeige- zeit	Bibliothek	Objekt	Objektart	Objekt- attribut
000001 *****	1 ***** En	230607 de des Ber	143422 ichts ***	BOEHMER	QRY001	*QRYDFN	QRY

#### Gefundene Abfragen



#### Benötigte Felder

Ich setze die in den vorhergehenden Abschnitten erstellten Klassen ein, um die Daten bereitzustellen. Die Ausgabe als HTML-Tabelle erfolgt analog zu den bisher gezeigten Sripten. Es ist also keine Entwicklung neuer Routinen notwendig. Es werden lediglich nicht alle Spalten der Datei angezeigt, sondern nur die beiden benötigten.





5.2.3 Seite 23

```
$0BJ = 'B0EHMER/*ALL';
$0BJTYPE = '*0RYDFN';
$0UTFUT = '*0UTFILE';
$0UTFILE = 'B0EHMER/FLEQRY';
$ListQueries = new I5Command( 'DSP0BJD' );
$ListQueries->addParm( new I5CommandParameter( '0BJ', $0BJ ));
$ListQueries->addParm( new I5CommandParameter( '0BJTYPE', $0BJTYPE ));
$ListQueries->addParm( new I5CommandParameter( '0UTPUT', $0UTPUT ));
$ListQueries->addParm( new I5CommandParameter( '0UTPUT', $0UTPUT ));
$ListQueries->addParm( new I5CommandParameter( '0UTFILE', $0UTFILE ));
$ListQueries->addParm( new I5CommandParameter( '0UTFILE', $0UTFILE ));
$done = $ListQueries->run( $Connection );
```

Bereitstellung der Daten

```
echo '';
echo ''.$arrDataRow[ 3 ].'';
echo ''.$arrDataRow[ 4 ].'';
echo '';
echo '';
```

HTML-Tabelle erzeugen

🖉 http://172.16.110.22/basics/script001_4.php - Windows Internet Explorer
(C) - (e) http://172.16.110.22/basics/script001_4.php
2 4 10 110.22/basics/script001_4.php
BOEHMER QRY001

Anzeige im Browser



5.2.3 Seite 24

# 5.2.3.1.9 Webfrontend mit Assistenten ausstatten.

Die auszuführende Abfrage soll aus einer Liste ausgewählt werden können. Die Anzeige der Liste verfügbarer Abfragen wird in einer neuen Seite angezeigt und der Benutzer kann dort nach Bibliotheken filtern, in denen nach Querydefinitionen gesucht werden soll. Zur Anzeige der Auswahlmöglichkeiten füge ich einen weiteren Submit-Button in die Befehlsoberfläche ein. Das Ganze sieht im Browser dann so aus.

Abfrage ausführen (RUN	IQRY ) - Windows Internet Explorer .110.22/basics/script003_1.php
😵 🌈 Abfrage ausführer	(RUNQRY)
Abfrage ausfü	ihren ( RUNORY )
Abfrage	suchen
Bibliothek	
Ausgabeform *DETAIL	
Ausgabedatei	
Datei	

Neuer Submit-Button im Browser



HTML-Code



Seite 25

Der Ablauf wird durch die Einführung weiterer Seiten komplexer. Darum werde ich den vorgesehenen Ablauf skizzieren. Der erste Gedanke gilt der letzten abgebildeten Seite, der Befehlsoberfläche. Nach der Auswahl eines auszuführenden Queries soll der Anwender wieder zu dieser Seite gelangen. In den Eingabefeldern für Abfrage und Bibliothek sollen die Werte stehen, die im Assistenten ausgewählt wurden. Beim ersten Aufruf der Seite sind diese Felder vorerst leer.

Daher wird die Erstellung der Eingabefelder nun von PHP übernommen. Der neue Code wird die <input...> – Elemente ausgeben und mit eventuell vorhandenen Daten aus dem \$\_POST-Array füllen. Ich hatte bereits erwähnt, dass die Inhalte von Steuerelementen eines Formulars in dieses Array geschrieben werden, wenn das Formular über einen Submit-Button abgesendet wird.

<pre>% pup ecno ~ &lt;</pre>	Input cype-("text("".
	"name=\"QRYNAME\" ".
Inputs als String	"maxlength=\"10\" ".
erstellen	"value=\"".\$_POST[ 'QRYNAME' ]."\"".
	"/> "; ?>
="s	ubmit" name="lstQry" value="suchen" />
Bibliothek <td>&gt;</td>	>
php echo " <</td <td>input type=\"text\" ".</td>	input type=\"text\" ".
	"name=\"QRYLIB\" ".
	"maxlength=\"10\" ".
	"value=\"".\$_POST[ 'QRYLIB' ]."\"".
	"/> ": 2>

Aufbauen der Eingabefelder

Wenn der Anwender auf den *ausführen*-Button klickt, läuft die Abfrage wie bisher ab und zeigt das Ergebnis auf einer neuen Seite.



5.2.3 Seite 26

#### 5.2.3.1.10 Anzeigen von Bibliotheken, die Queries enthalten

Der erste Teil des Assistenten soll Bibliotheken in einer Tabelle anzeigen, die eine oder mehrere Abfragen enthalten. Die gewünschte Bibliothek soll durch einen Radiobutton auswählbar sein und die erste Bibliothek (falls vorhanden) soll vorselektiert sein. Das Script wird den Namen *libList.php* erhalten und eine Abwandlung des Code von Abschnitt 5.2.3.1.6 sein. Dort wurden ja die Abfragen **einer** Bibliothek angezeigt. Ich stelle Ihnen in der nächsten Abbildung die Ansicht im Browser vor.

C http://	/172.16.110.22/basics/libList.php - Windows Internet Explorer
00	<ul> <li>Image: bitp://172.16.110.22/basics/libList.php</li> </ul>
* *	6 http://172.16.110.22/basics/libList.php
Auswahl	Bibliothek
۲	BOEHMER
0	EPSSVFIL
0	EPSW02
0	EPSW0201
0	EPSW0202
0	EPS001CR
0	EPS004CR
0	QSYS
Que	eries anzeigen

Bibliotheken, die Abfragen enthalten

Wenn das Script gestartet wird, werden die Daten über einen DSPOBJD-Befehl in der Datei FLEQRY in der Bibliothek BOEHMER bereitgestellt.

```
// Bereitstellung aller Bibliotheken, die Abfragen enthalten.
$OBJ = '*ALL/*ALL';
$OBJTYPE = '*QRYDFN';
$OUTFUT = '*OUTFILE';
$OUTFILE = 'BOEHMER/FLEQRY';
$ListQueries = new I5Command( 'DSPOBJD' );
$ListQueries->addParm( new I5CommandParameter( 'OBJ', $OBJ ) );
$ListQueries->addParm( new I5CommandParameter( 'OBJTYPE', $OBJTYPE ) );
$ListQueries->addParm( new I5CommandParameter( 'OUTPUT', $OUTFUT ) );
$ListQueries->addParm( new I5CommandParameter( 'OUTPUT', $OUTPUT ) );
$ListQueries->addParm( new I5CommandParameter( 'OUTFILE', $OUTFILE ) );
$done = $ListQueries->run( $Connection );
```



Über die bekannte *odbc\_connect*-Funktion wird eine Datenbankverbindung in die PHP-Variable \$ODBC\_Cnc gestellt.

```
$ODBC_Cnc = odbc_pconnect( "EPOSI501", "BOEHMER", "******");
if( !$ODBC_Cnc ) echo odbc_errormsg();
```

Aus dieser Datei wird über ein SQL-Select die Liste der Bibliotheken abgefragt. Verwendet wird nur die Spalte, die den Namen der Bibliothek enthält, der ja letztendlich in der Tabelle ausgegeben werden soll.

```
$strSelect = "SELECT DISTINCT ODLENM FROM BOEHMER.FLEQRY";
$rsLibraries = odbc_exec( $ODBC_Cnc, $strSelect );
```

# SQL-Statement erstellen und Daten abrufen

Die gelieferten Sätze finden sich in dem ResultSet \$rsLibraries. Anschließend folgt Code, der die Sätze aus dem ResultSet ausliest und jede Bibliothek mit einem Radiobutton in einer Tabellenzeile ausgibt. Die jetzt vorgestellten Codezeilen sind zusätzlich in einem Formular eingebettet, das nachher noch beschrieben wird. Zuerst aber zu der Tabellenerstellung, beginnend mit Spaltenüberschriften.

```
echo "";
    // Spaltenheader ausgeben
    echo "";
        echo ">";
        echo ">"."Auswahl"."";
        echo ">"."Bibliothek"."";
        echo "
```

# Header

Bevor die Schleife zur Ausgabe einer Zeile anläuft, wird eine PHP-Variable \$first auf true gesetzt. Innerhalb der Schleife wird bei \$first == true ein selektierter Radiobutton dargestellt und anschließend \$first = false gesetzt. Damit ist sichergestellt, dass die erste Bibliothek der Liste wie gewünscht vorselektiert ist.





Radiobuttons erstellen

}

\$first = false;

In diesen Zeilen fällt die Variable \$Lib auf. Sie enthält den Namen der Bibliothek des aktuell abgerufenen Satzes des ResultSets und wird mit diesem Code ermittelt.

```
$Lib = odbc_result( $rsLibraries, 1 ); // Name der Bibliothek
```





Betrachten wir den Code einmal im Zusammenhang. Kasten (1) zeigt die Vorbelegung der Toggle-Variablen \$first und die Variable \$continue, die als Anzeiger für die Schleifenprüfung dient. Ihr Wert wird der Rückgabe des Befehls *odbc\_ fetch...* entnommen, der den nächsten Satz aus dem ResultSet abruft. Siehe auch (3). Kasten (2) zeigt die Umschaltung der Toggle-Variablen und (4) fügt den Submit-Button für die Auflistung der Anfragen der selektierten Bibliothek hinzu.

Nun fehlt nur noch das Formular. Hier wird dafür Sorge getragen, dass das Script qryList.php mit der Methode POST aufgerufen wird, wenn ein Submit-Button angeklickt wird. Dieses Script wird dafür verantwortlich sein, die Abfragen darzustellen.



# 5.2.3.1.11 Anzeigen der Abfragen in einer Bibliothek

Aus dem \$\_POST-Array kann der Parameter \$\_POST[ 'QRYLIB' ] entnommen werden, wenn qryList.php wie im letzten Abschnitt beschrieben über die Methode POST aufgerufen wird. Der Inhalt dieses Elementes wird verwendet, um auf der Seite die Information, auf welche Bibliothek man sich gerade bezieht, auszugeben. Außerdem wird auch in dieser Seite ein SQL-Select verwendet, um die Namen der Abfragen zu ermitteln. Dieses Statement muss natürlich auch nach der gewünschten Bibliothek selektieren.

```
$strSelect = "SELECT ODOBNM FROM BOEHMER.FLEQRY WHERE ODLENM = '". POST[ 'QRYLIB' ]."";
$rsQueries = odbc_exec( $ODBC_Cnc, $strSelect );
```

# Abfragenamen ermitteln

Auch in diesem Kapitel zeige ich Ihnen zuerst das Aussehen der Webseite im Browser, bevor ich den zugrundeliegenden Code erläutern werde.



Liste der Abfragen im Browser



Web-Programmierung

PHP und i5/OS-Befehle

5.2.3 Seite 30 Die Dinge sind in einem Formular eingebettet und das Attribut *action* sorgt dafür, dass via Submit-Button wieder der Befehlsprompt für die Ausführung eines Queries aufgerufen wird. Die Inhalte werden wieder mit *POST* geliefert.

In der letzten Zeile erkennt man, wie Informationen über die durchsuchte Bibliothek ausgegeben werden. Der Inhalt des Feldes mit dem Namen *QRYLIB* wird mit dem Inhalt des gleichnamigen \$\_POST-Arrayelements besetzt. Da es sich hier um ein Eingabefeld handelt, wird dieser Inhalt beim Submit auch an den Befehlsprompt übergeben.

Dann werden wieder die Spaltenüberschriften der Tabelle hergestellt.

```
echo "";
    // Spaltenheader ausgeben
    echo "";
        echo ""."Auswahl"."";
        echo ""."Abfrage"."";
        echo "
```

# Spaltenüberschriften

Der anschließende Code ist mit dem Quellcode für die Darstellung der Bibliothekenliste vergleichbar. Der ausgewählte Abfragename wird auch mit dem \$\_POST-Array übergeben. Da der Inhalt eine so weitgehende Ähnlichkeit mit dem des vorhergehenden Abschnitts aufweist, verzichte ich auf eine weitere Besprechung.





Seite 31



Tabelle der Abfragenamen aufbauen

Zum Abschluss des Scripts findet sich die Ausgabe des notwendigen Buttons, der wieder zum Befehlsprompt zurückführt.

Noch einmal: Die Inhalte des Eingabefelds mit dem Namen der aktuellen Bibliothek und der Wert des selektierten Radiobuttons stehen über das \$\_POST-Array dem Script runQry.php zur Verfügung!





# 5.2.3 Seite 32

#### 5.2.3.1.12 Übernahme der Assistenten-Daten in den Befehlsprompt

Wenn das Script runQry.php wieder aufgerufen wird, müssen die übergebenen Daten aus dem \$\_POST-Array in die entsprechenden Eingabefelder geschrieben werden. In diesem Script muss außerdem eine Entscheidung getroffen werden, welche Aktion durchgeführt werden soll.

DUNODV
RUNQRY)
en

So sieht die Anzeige aus, wenn die Seite das erste Mal aufgerufen wurde. Der Anwender kann nun die einzelnen Felder ausfüllen und auf *ausführen*... klicken. Das ist der ganz zu Beginn beschriebene Weg. Variante 2 ist der Klick auf *suchen*... ; Resultat: die Liste der Bibliotheken wird angezeigt.

<body></body>	
<form< td=""><td>id="form1" action="runQry.php" method="POST"&gt;</td></form<>	id="form1" action="runQry.php" method="POST">
<h< td=""><td>1&gt;Abfrage ausf<mark>ü</mark>hren ( RUNQRY )</td></h<>	1>Abfrage ausf <mark>ü</mark> hren ( RUNQRY )

Action bei Klick auf ausführen...

Klickt der User auf *ausführen*..., so wird die Seite selbst aufgerufen. Ob ein solcher Aufruf stattgefunden hat, ist zu Beginn der Seite zu testen. Hier der zugeordnete Code gleich am Anfang der Seite.





5.2.3 Seite 33

```
<?php
if( $_POST[ 'Submit' ] == 'suchen...' ){
    header("Location: libList.php");
}
if( $_POST[ 'Submit' ] == 'ausführen...' ){
    $getparms = "?QRYLIB=".$_POST[ 'QRYLIB' ].
        "&QRYNAME=".$_POST[ 'QRYNAME' ].
        "&QRYNAME=".$_POST[ 'QRYNAME' ].
        "&GUTFORM=".$_POST[ 'OUTFORM' ].
        "&FLELIB=".$_POST[ 'FLELIB' ].
        "&FLENAME=".$_POST[ 'FLELIB' ].
        "&FLENAME=".$_POST[ 'FLELIB' ];
        $header = "Location: script001_3.php".$getparms;
        header( $header );
}
</pre>
```

Was geschieht hier?

Wenn auf den *suchen*-Button geklickt wurde, steht in \$\_POST[ 'Submit' ] der Wert "*suchen*...". Das bedeutet ja, dass die Liste der Bibliotheken angezeigt werden soll, die mindestens eine Abfrage enthalten. Die Weiterleitung auf die Seite geschieht über die header(...)-Anweisung.

Wenn nicht *suchen*... enthalten ist, wird untersucht, ob *ausführen*... angeklickt wurde (nach dem gleichen Prinzip). In diesem Fall wird eine Liste von get-Parametern aufgebaut, in \$getparms abgelegt und dann der Weiterleitungs-String erstellt. Die Behandlung der get-Parameter in einer Seite erfolgt analog zur POST-Methode. Weitergeleitet wird in jedem Fall dann zu dem Script, das die Abfrage durchführt und das Ergebnis dann als Tabelle darstellt. Wenn Sie den Ablauf nicht mehr ganz im Gedächtnis haben, schauen Sie ruhig noch einmal am Beginn des ganzen Kapitels nach. Dort hatte ich diese Funktionalität aufgebaut.

Sollte keiner der beiden Zweige in Frage kommen, handelt es sich um den ersten Aufruf der Seite. Oh je, dieses Kapitel war schon nicht mehr ganz so trivial, aber wenn man noch ein wenig Energie in das Design legt und den Code weiter strukturiert... hat man etwas für den praktischen Gebrauch.

Zum Abschluss nun noch eine Bildfolge durch den Assistenten, nachdem auf *suchen...* geklickt wurde.



Kapitel 2

# Web-Programmierung

PHP und i5/OS-Befehle

# 5.2.3 Seite 34

)6	> • 🛃 http://172.16.110.22/basics/libList.php	
r 44	C http://172.16.110.22/basics/libList.php	
Auswa	ahl Bibliothek	
$\odot$	BOEHMER	
0	EPSSVFIL	
0	EPSW02	
0	EPSW0201	
0	EPSW0202	
0	EPS001CR	
0	EPS004CR	
0	OSVS	







5.2.3 Seite 35







Kapitel

**Web-Programmierung** PHP und i5/OS-Befehle

5.2.3 Seite 36





# 5.2.4 DB2-Funktionen

Beginnen wir jetzt mit der Vorbereitung zur Programmierung eines PHP-Scripts, welches mit Hilfe der DB2-Funktionen des PHPs Daten tabellarisch anzeigt.

Diese nachfolgend gezeigten Arbeitsschritte sollten Sie auf Ihren Systemen nachvollziehen.

- 1. Ich beabsichtige alle restlichen PHP-Programmierschritte im IFS-Verzeichnis /www/zendcore/htdocs/phponsystemi zu realisieren. Deshalb richte ich mir eine schreib-/lesefähige Freigabe mit dem System i Navigator auf dem Verzeichnis phponsystemi ein.
- 2. Dieser Freigabe ordne ich auf meinem PC ein Netzwerklaufwerk zu.
- 3. Im freigegebenen Verzeichnis phponsystemi erstelle ich die Unterverzeichnisse Styles, Classes, Includes.
  - Styles beinhaltet später meine CSS-Dokumente.
  - Classes beinhaltet später von mir zu erstellende PHP-Klassen.
  - Includes wird später Copy-Strecken enthalten.
- 4. Ich rufe dann Zend Studio auf und erzeuge hier ein Projekt.



Neues Projekt erstellen





# Web-Programmierung

DB2-Funktionen



5. Ich vergebe den Projektnamen PHPOnSystemi.



Projektname vergeben

Ich speichere das Projekt in einem eigenen Pfad auf meinem PC und drücke "Weiter".

6. Ich gebe das Laufwerk z: als Pfad an, in dem sich meine PHP-relevanten Dokumente befinden.







	Web-Programmierung	
	DB2-Funktionen	
Nachdem ich mit "Beenden" den Projekterstellungsdiale	og beendet habe, sieht	5.2.4
die Projektsicht im Zend Studio wie folgt aus.		Seite 3
📔 PHPOnSystemi - Zend Development Environment - [PHPDoc	11	
<u>D</u> atei Bearbeite <u>n</u> <u>S</u> uchen <u>G</u> ehe zu <u>P</u> rojekt <u>A</u> nsicht De <u>b</u> ug <u>W</u> erkzeug		
📉 🛛 🞾 Dateisystem 🛛 🗁 Projekt  🧐 SQL 🛛 🖕		
p Projekt: PHPOnSystemi		
phponsystemi auf "172.16.111.1" (Z;)      Classes     Fri Includes     Fri Styles     Tabelle.html		

Projekt: PHPOnSystemi

#### 5.2.4.1 Die Aufgabe

All Files

In unserem ersten Projektabschnitt beabsichtige ich, eine tabellarische Anzeige von Kundendaten zu realisieren.

Dazu müssen Sie wissen, dass die Kundendaten in der Datei SVKLST00 zu finden sind. Da die Datei SVKLST00 jedoch auch Lieferantendaten beinhaltet, müssen wir diese beiden Bereiche voneinander abgrenzen. Abgrenzungsbegriff ist die ID-Nummer (KLIDNR).

- KLIDNR between 1.000.000 and 6.999.999 entspricht Kunden.
- KLIDNR > 7.000.000 entspricht Lieferanten.

Folgende Felder der Datei sollen angezeigt werden:

- KLIDNR: ID-Nummer,
- KLNAME: Name,
- KLLANDKZ: Land,
- KLSTR: Strasse,
- KLORT: Ort,
- KLTEL: Telefon,
- KLDATN: Datum der Erfassung.





# Web-Programmierung

DB2-Funktionen

Mit folgendem SQL-Befehl sollen die Daten ausgelesen werden:

5.2.4 Seite 4

select
klidnr,
klname,
(select slbez from epsw0402.svslst00
where slkey = ,LAND' and slwert = kl.kllandkz) as klland,
klstr,
klort,
kltel,
kltel,
kldatn
from epsw0402.svklst00 kl
where klidnr between 1000000 and 6999999
order by klidnr

Eine Besonderheit dieser Abfrage ist, dass das Feld KLLANDKZ, welches das Länderkennzeichen beinhaltet, durch ein scalares Subselect auf die Datei SVSLST00 – Schlüsselstamm in die Länderbezeichnung übersetzt wird.

Wir testen jetzt diese SQL-Abfrage auf Funktionsfähigkeit.

Öffnen Sie dazu in Ihrem Zend Studio die Sicht SQL.



SQL-Einstellungen



			Wel	o-Programmierung	ia ia
				DB2-Funktionen	Ka
Fordern Sie mit der	rechten Maustaste	e den Dialog "I	Einstellungen"	an.	5.2.4
Systemi Einstellung	jen		×		Seite 5
SQL-Server-Einstellur	igen				
Server-Typ:	DB2/400	<b>V</b>			
Server-Name (Alias)	Bystemi				
Hostname/IP:	172.16.111.1				
Port:	8475				
Datenbankname:	ITPSERVER				
Benutzername:	LUTTKUS				

Hilfe

#### Verbindungsdaten

Passwort:

Verbindungs-Einstellungen

rbindun

Z Auto-Reconnect

🗌 Teste dig

OK

Tragen Sie in den Dialog die für Ihr System relevanten Verbindungsdaten ein. Testen Sie die Verbindung. Beenden Sie den Dialog mit "OK".

Sekunden

Abbrechen

Fordern Sie das Kontextmenü für System i an.

\* \* \* \* \* \*

le

Test

🔽 Verbinde wenn Zend Studio startet 🛛 🔽 Passwort speichern

10 🗧

0

Web-Programmierung

DB2-Funktionen

apite



Daten abfragen

Übernehmen Sie die SQL-Abfrage in die Sicht SQL und klicken Sie auf das Ausführungssymbol.

Im zentralen Bereich des Zend Studios sehen Sie jetzt das Ergebnis der ausgeführten Abfrage.

Gleichzeitig haben Sie gelernt, dass das Zend Studio ein sehr guter DB2 SQL-Client ist, indem Sie die SQL-Abfragen, die Sie in PHP-Scripten verwenden wollen, testen können.





Seite 7

ebnis	sse auf Seite:	100 -			2 🔶 🕯
	KLIDNR	KLNAME	KLLAND	KLSTR	KLOF
1	1000200	BT Deutschland	Deutschland	 Grovestr. 16	Langenhage
1	1000500	Coty Deutschla	Deutschland	 Rheinstr. 4e	Mainz
1	1001200	ADP Employer S	Deutschland	 Im Hollergrund	Bremen
1	1001500	Händle GmbH &	Deutschland	 Industriestr	Mühlacker
1	1001700	F.W. Hempel &	Deutschland	 Leopoldstraße	Düsseldori
1	1001900	Unilog Integra	Deutschland	 Schleifmühlewe	München
1	1002200	Lloyd Shoes Gm	Deutschland	 Hans-Hermann-M	Sulingen
1	1002300	Kaugummifabrik	Deutschland	 Paul-Ehrlich-S	Rödermark
1	1002800	Pracht Spediti	Deutschland	 Hansastraße 10	Haiger
1	1003100	CIBA Vision Ve	Deutschland	 Industriering	Großwallst
1	1003300	Beeck Feinkost	Deutschland	 Albert-Schweit	Hamburg
1	1003400	Hoyer GmbH Int	Deutschland	 Wendenstr. 414	Hamburg
1	1004300	Rotes Kreuz Kr	Deutschland	 StPauli-Deic	Bremen
1	1004400	Lloyd Dynamowe	Deutschland	 Hastedter Oste	Bremen
1	1004700	Certuss-Warent	Deutschland	 Hafenstr. 65	Krefeld
1	1004800	Vereinigte Bür	Deutschland	 Schwarzwaldstr	Tottnau
1	1006200	OPITEC Handel	Deutschland	 Hohlweg 1	Giebelstad
1	1006400	IHK Zentrum fü	Deutschland	 Salomon-Idler	Augsburg
1	1006900	SWING Software	Deutschland	 Bremer Str. 59	Achim
1	1007100	Carl Wilhelm M	Deutschland	 Ammerländer He	Oldenburg
1	1007200	Wächter & Wäch	Deutschland	 Außer der Schl	Bremen
1	1007700	WITT Gasetechn	Deutschland	 Salinger Feld	Witten
1	1007900	UBS Private Ba	Deutschland	 Stephanstr. 14	Frankfurt
1	1008200	RICOH DEUTSCHL	Deutschland	 Mergenthaler A	Eschborn
1	1009000	KARL KÖNECKE G	Deutschland	 Sebaldsbrücker	Bremen
1	1009600	Baecker AG	Deutschland	 Grauwallring 1	Bremerhave
1	1009800	Bindan GmbH &	Deutschland	 Bahnhofstr. 8	Stuhr
					•

#### Abfrage-Ergebnis

Sie sehen, dass die Länderkennzeichen erfasst übersetzt wurden.

# 5.2.4.2 Benötigte DB2-Funktionen

Neben den bereits verwendeten ODBC-Funktionen kennt PHP noch weitere Datenbank-spezifische Funktionen, die über native Schnittstellen auf die jeweilige Datenbank zugreifen können. Seit Zendcore für i5/OS kann die db-Funktionsfamilie voll funktional zusammen mit DB2 für System i angewendet werden.

# 5.2.4.2.1 Db2\_connect()

Db2\_connect – Baut die Verbindung zur DB2.

int db2\_connect ( string dsn, string user, string password [, options] )

Liefert eine ODBC-Verbindungskennung connection\_id oder 0 (FALSE) bei Fehlfunktion zurück.

Die Verbindungskennung connection\_id wird von den anderen DB2-Funktionen benötigt. Es können mehrere DB2-Verbindungen nebeneinander bestehen.



# 5.2.4 5.2.4.2.2 Seite 8 db2\_exec()

db2\_exec - Bereitet einen SQL-Befehl auf und führt ihn aus.

int db2\_exec ( int connection\_id, string query\_string )

Liefert bei einem Fehler FALSE zurück, sonst eine DB2-Ergebniskennung result\_id.

**Db2\_exec**() – Sendet einen SQL-Befehl zu dem Datenbankserver, der durch die Verbindungskennung connection\_id bezeichnet wird. Dieser Parameter muss durch **db2\_connect**() erzeugt worden sein.

# 5.2.4.2.3 db2\_fetch\_row()

db2\_fetch\_row – Liefert eine Datenzeile zurück.

int db2\_fetch\_row ( int result\_id [, int row\_number] )

Wenn **db2\_fetch\_row()** erfolgreich durchgeführt wurde (also eine oder mehrere Datenzeilen gefunden wurden), dann wird **TRUE** zurückgeliefert. Wenn keine (weitere) Zeile vorhanden ist, wird **FALSE** zurückgegeben.

**db2\_fetch\_row**() stellt eine Zeile der von **db2\_exec**() zurückgegebenen Daten bereit. Nachdem **db2\_fetch\_row**() aufgerufen wurde, können die Felder dieser Zeile mit **db2\_result**() genutzt werden.

Wenn row\_number nicht definiert ist, versucht **db2\_fetch\_row**() die nächste Zeile aus dem Abfrageergebnis result\_id zu holen. **Db2\_fetch\_row**() kann abwechselnd ohne und mit dem Parameter row\_number genutzt werden.

Um ein Abfrage-Ergebnis mehrfach zu durchlaufen, kann man **db2\_fetch\_ row**() mit row\_number gleich 1 aufrufen, um dann wie gewohnt mit **db2\_ fetch\_row**() ohne Zeilenparameter fortzufahren.





#### 5.2.4.2.4 db2\_result()

db2\_result – Erlaubt den Zugriff auf die Ergebnisdaten.

string db2\_result ( int result\_id, mixed field )

Liefert den Inhalt eines bestimmten Feldes zurück.

Der Feldparameter field kann entweder der Integerwert der Spaltennummer oder ein String mit dem Spaltennamen sein.

Als Beispiel:

\$produkt\_3 = db2\_result (\$Query\_ID, 3);

\$produkt\_wert = db2\_result (\$Query\_ID, ",wert");

Der erste Aufruf von **db2\_result**() liefert den Wert des dritten Feldes im aktuellen Datensatz im Abfrage-Ergebnis zurück. Der zweite Funktionsaufruf gibt den Wert der Spalte "Wert" im aktuellen Datensatz zurück. Wenn die Spaltennummer kleiner als 1 oder größer als die Spaltenzahl ist, tritt ein Fehler auf. Gleiches gilt für einen Spaltennamen, der im Abfrageergebnis nicht vorkommt.

Die Spaltennummern beginnen mit 0.

# 5.2.4.2.5 db2\_close()

db2\_close - Beendet eine ODBC-Verbindung.

void db2\_close ( int connection\_id )

**db2\_close**() beendet die Verbindung zum Datenbankserver, die durch das Verbindungskennzeichen connection\_id gekennzeichnet wird.

# Anmerkung:

Diese Funktion scheitert, wenn auf dieser Datenbankverbindung noch Transaktionen ausstehen. In diesem Fall bleibt die Verbindung bestehen.





5.2.4 Seite 10

#### 5.2.4.3 Das Script SVKLST00.PHP

Wir werden jetzt das erste Script realisieren, in dem wir die abgefragten Kundendaten in einer Tabelle anzeigen. Die Abfrage wird mit Hilfe der PHP DB2-Funktionen ausgeführt.

Ich erstelle mit Hilfe des Zend Studios eine neue Datei: svklst00.php



Neue Datei erstellen

- 1. Wählen Sie die Sicht "Projekt" an.
- 2. Rechte Maustaste → Phponsystemi... → Neue Datei







<html> <head></head></html>
<pre>charset=iso-8859-1" /&gt; <title>Kunden und Lieferantenliste</title></pre>
<body></body>
php</td
<pre>\$system = "ITPSERVER"; \$user = "WWWUSER"; \$passwd = "INTERNET";</pre>
<pre>\$connectionstring = db2_connect(\$system,</pre>

DB2-Funktionen



5.2.4 Seite 11




DB2-Funktionen



\$klname = db2\_result(\$queryexe, ,KLNAME'); \$klland = db2\_result(\$queryexe, 'KLLAND'); \$klplz = db2\_result(\$queryexe, 'KLPLZ'); \$klort = db2\_result(\$queryexe, 'KLORT'); \$klstr = db2\_result(\$queryexe, 'KLSTR'); \$kltel = db2\_result(\$queryexe, 'KLTEL'); \$kldatn = db2\_result(\$queryexe, 'KLDATN'); print(,,<TR>"); print(,,<TD>\$klidnr</TD>"); print(,,<TD>\$klname</TD>"); print(,,<TD>\$klland</TD>"); print(,,<TD>\$klplz </TD>"); print(,,<TD>\$klort </TD>"); print(,,<TD>\$klstr </TD>"); print(,,<TD>\$kltel </TD>"); print(,,<TD>\$kldatn</TD>"); print(,,</TR>"); } print(,,</TBODY>"); print(,,</TABLE>"); db2\_close(\$connectionstring); ?> </body> </html>

Seite 13

Sehen wir uns einmal dieses Script an. Es ist Ihnen sicherlich aufgefallen, dass ich die meisten Bestandteile in reinem PHP ausgeführt habe. Ich habe mich zu diesem Schritt entschieden, da ich nicht allzu viele HTML- und PHP-Codeinseln erzeugen möchte.





# 5.2.4 Seite 14

# Hier die Ausgabe:

Identnr. Name	Land	PLZ	Ort	Strasse	Telefon	Kunde seit
1014500 Kraft Foods Deutschland GmbH & Co. KG	Deutschland	28199	Bremen	Langemarckstr. 4-20	0421/59901 e	2002-12-03
1027400 Harrods Knightsbridge Limited	Groß Britannien	SWIX 7X	London	Knightsbridge	0044/1717301234	2002-05-29
1042000 Berendsohn AG	Deutschland	22559	Hamburg	Alte Sülldorfer Landstr. 400	040/81909106 e	2002-05-29
1064000 JD Edwards Deutschland GmbH	Deutschland	64546	Mörfelden/Frankfurt	Waldeckerstrasse 11	06103/7260 e	2002-12-11
1089900 Wort+Bild Verlag Konradshöhe GmbH & Co.	Deutschland	82065	Baierbrunn	Konradshöhe	089/744330 e	2002-12-12
1142600 ADS Informatik Hard- u. Software-Beratung	Deutschland	66693	Mettlach	Goetheweg 8	06865/93300 e	2002-12-12
1202600 Smith + Nephew Richards GmbH Medizintechnik	Deutschland	78532	Tuttlingen	Alemannenstr. 14	07462/2080 e	2003-02-11
1205800 DIFA Deutsche Immobilien Fonds AG	Deutschland	20354	Hamburg	Valentinskamp 20	040/349190 e	2002-05-29
1224500 HUDSON Intern. Vertr. GmbH	Deutschland	70565	Stuttgart	Am Wallgraben 45	0711/78150	2002-05-29
1231200 Holert Kalksandsteinwerke GmbH & Co.	Deutschland	24568	Kaltenkirchen	Barmstedter Str. 14	04191/7050 e	2002-05-29
1257600 Fonds f. Umweltstudien e.V.	Deutschland	53175	Bonn	Godesberger Allee 108 - 112	0228/2692/0	2002-05-29
1272200 IDS Inter Data	Deutschland	63477	Maintal	Am Kreuzstein 82-84	06109/76680	2002-05-29
1311600 TDS Tele Columbus Daten und Service GmbH	Deutschland	30519	Hannover	Peiner Str. 8	0511/98765 e	2002-05-29
1317100 Virgin Records GmbH	Deutschland	80803	München	Herzogstr. 64	089/381950	2002-05-29
1317200 Gutehoffnungshütte Radsatz GmbH	Deutschland	46145	Oberhausen	Steinbrinkstr. 170	0208/7400/0 e	2002-05-29
1335000 DSM Computer	Deutschland	57299	Burbach	Forstweg 14	02736/3176	2003-04-22
1352900 IDS GmbH	Deutschland	69517	Gorxheimertal	Am Wetzelsberg 2	06201/292091 e	2003-04-15
1353200 Berendsen PMC GmbH Steinebronn Industriebedarf Geschäftsbereich	Deutschland	70499	Stuttgart	Kranstr. 9	0711/8361/0	2002-05-29
1377900 Verbandsgemeindeverwaltung EDV-Organisation	Deutschland	57610	Altenkirchen	Rathausstraße 13	02681/850	2002-05-29
1380100 Harley-Davidson GmbH	Deutschland	64546	Mörfelden-Walldorf	Starkenburgstr. 12	06105/2840	2002-05-29
1385000 Feige, Siegfried LKW-Instandsetzung	Deutschland	02708	Löbau	Weißenberger Str. 45	03585/862537 e	2002-10-09
1434300 Mundschenk Druck- und Verlags- gesellschaft mbH	Deutschland	29614	Soltau	Harburger Str. 63	05191/808/0 e	2002-02-18
1464700 Unilever Best Foods Deutschland GmbH	Deutschland	20355	Hamburg	Dammtorwall 15	040/3493/0 e	2002-02-06
1483200 HDS Rechenzentrum GmbH & Co. KG	Deutschland	49577	Ankum	Am Lordsee 1	05462/765/0 e	2002-08-05
1498400 Randstad Organisation für Zeitarbeit GmbH	Deutschland	65760	Eschborn	Ginheimerstr. 5	06196/408/0 e	2003-01-09
2032801 Deutsche Immobilien Fonds AG DIFA	Deutschland	20355	Hamburg	Caffamacherreihe 8	040/34919-0 e	2003-01-14
2045201 WestInvest Gesellschaft für Investmentfonds mbH	Deutschland	40476	Düsseldorf	Hans-Böckler-Str. 33	0211/51798-0 e	2002-06-05
2054401 Fonds für Umweltstudien e.V.	Deutschland	53175	Bonn	Godesbergerallee 108-112	0228/2692216 e	2003-06-02
2054701 EDS Informationstechnologie & Service GmbH	Deutschland	53229	Bonn	Kautexstr. 52	0228/488-0 e	2002-11-06

Ausgabe svklst00.php





# 5.2.5 Dialoge erstellen

Nachdem ich Ihnen im vorherigen Abschnitt Daten in tabellarischer Form gezeigt habe, geht es in diesem Kapitel darum, Daten einzugeben, die anschließend vom Server weiterverarbeitet werden können.

HTML bezeichnet solche Dokumente als Formulare, in denen man Daten eingeben kann. Der Anwender kann in Formularen "Eingabefelder" ausfüllen, in mehrzeiligen Textfeldern "Text" eingeben oder aus Listen "Einträge" auswählen. Wenn das Formular fertig ausgefüllt ist, kann er auf einen Button klicken, um das Formular abzusenden.

Beim Erstellen eines Formulars geben Sie an, was mit den Daten des ausgefüllten Formulars passieren soll. Sie können die Daten dann von einem PHP-Script auf dem Server-Rechner weiterverarbeiten lassen.

Formulare können sehr unterschiedliche Aufgaben haben. So werden sie zum Beispiel eingesetzt:

- um bestimmte, gleichartig strukturierte Auskünfte von Anwendern einzuholen,
- um Anwendern das Suchen in Datenbeständen zu ermöglichen,
- um Anwendern die Möglichkeit zu geben, selbst Daten für einen Datenbestand beizusteuern,
- um dem Anwender die Möglichkeit individueller Interaktion zu bieten, etwa um aus einer Produktpalette einen bestimmten Artikel zu bestellen.

Ein Software-Hersteller könnte beispielsweise ein Formular zur Verfügung stellen, in dem der Anwender angeben kann, welche Firmenprodukte er von ihm bereits besitzt, wie er von den Produkten erfahren hat, welchen Beruf er ausübt, auf welchem Rechnertyp die Software bei ihm läuft usw. Auf diese Weise kann ein Hersteller ein gleichartig strukturiertes und daher gut vergleichbares Feedback von den Anwendern einholen.

# 5.2.5.1 Die Aufgabe

Die Aufgabe dieses Kapitels besteht darin, einen Suchdialog zu realisieren, der es möglich macht, Kunden- und Lieferantendaten nach unterschiedlichen Kriterien zu durchsuchen. Das Ergebnis der Suche soll in tabellarischer Form angezeigt werden.



Dialoge erstellen

5.2.5 Seite 2 Die Selektionsdaten werden dann in ein Formular eingegeben. Anschließend werden die eingegebenen Daten aus dem Formular als Parameter an das tabellenerzeugende PHP-Script übergeben.

#### 5.2.5.2 Exkurs: HTML-Formulare

Sie können an einer beliebigen Stelle innerhalb des Dateikörpers einer HTML-Datei ein Formular definieren. Dazu müssen Sie im HTML-Dokument einen Formularbereich definieren.

```
<form action="svklst00.php " method="post">
<!-- hier folgen die Formularelemente -->
</form>
```

Mit "<form>...</form>" definieren Sie ein Formular. Alles, was zwischen dem einleitenden Tag "<form>" und dem abschließenden Tag "</form>" steht, gehört zum Formular. Das sind hauptsächlich Elemente des Formulars – wie Eingabefelder, Auswahllisten oder Buttons.

Um die Formularelemente zu platzieren und zu beschriften, können Sie dazwischen aber auch andere HTML-Elemente notieren. Dabei müssen Sie Folgendes beachten: Sie dürfen innerhalb eines Formulars nur Block-Elemente (und Script-Bereiche) notieren – also Überschriften, Textabsätze, allgemeine Bereiche oder Tabellen.

Im einleitenden Tag "<form>" geben Sie mit dem Pflichtattribut "action" an, an welches URI die Formulardaten beim Absenden des Formular übertragen werden sollen ("action" = Aktion). URI sollte die Adresse eines Prozesses auf dem Server-Rechner sein, das die Formulardaten weiterverarbeitet. Üblicherweise handelt es sich dabei um ein CGI-Script, das zum Beispiel in Perl und PHP geschrieben wurde.

Eine weiteres wichtiges Attribut bei der Formulardefinition ist das Attribut "method". Dabei bestimmen Sie, nach welcher HTTP-Übertragungsmethode die Formulardaten an ihr Ziel gelangen.

Hier gibt es zwei mögliche Werte: Wenn Sie "method="get"" wählen, hängt der Browser die Formulardaten als Parameter an die Aufrufadresse an (diese Angabe ist nicht zwingend erforderlich, da "get" als Default-Einstellung definiert ist). Beachten Sie, dass die Anzahl der übergebenen Zeichen je nach Browser auf zirka 2.000 limitiert sein kann.



Dialoge erstellen

Die Anfrage, die beim Server eintrifft, sieht dann beispielsweise wie folgt aus:

http://www.example.org/cgi-bin/feedback.cgi?AnwenderName=Stef an+M%FCnz&AnwenderMail=beispiel@example.org&Text=Das+ist+ein+ kleiner+Text.

Das Programm kann dies als parameterübergebene Zeichenkette verarbeiten.

Wenn Sie "method="post"" wählen, überträgt der Web-Browser die Formulardaten mit einer speziellen POST-Anfrage an den Web-Server ("post" = verschicken). Der Web-Server stellt die Daten über den Standardeingabekanal dem Programm zur Verfügung. Bei CGI-Scripts müssen Sie diese "POST"-Methode dann verwenden, wenn die Formulardaten zu umfangreich für die "GET"-Methode sind.

Das W3-Konsortium empfiehlt, die Methode "get" zu wählen, wenn das auswertende Programm die Daten zur Ablaufsteuerung benötigt (zum Beispiel für eine Suche oder zum Weiterblättern). Die Methode "post" wird dann empfohlen, wenn Daten über das auswertende Programm hinaus weiterverarbeitet werden (beispielsweise bei der Speicherung in einer Datenbank oder bei Auslösung einer Bestellung).

#### 5.2.5.2.1 Formularelemente

Innerhalb des Formularbereichs werden die Eingabeelemente definiert.

#### 5.2.5.2.1.1 Einzeilige Eingabefelder

```
<form action="svklst00.php " method="post">
Vorname : <input name="vorname" type="text" size="30"
maxlength="30"> value="Max"
Zuname : <input name="zuname" type="text" size="30"
maxlength="40" value="Meier">
</form>
```

"<input>" definiert ein einzeiliges Eingabefeld ("<input>"= Eingabe). Der Vollständigkeit halber sollten Sie die Angabe "type="text"" hinzusetzen.

Jedes Eingabefeld sollte einen internen Bezeichnernamen erhalten – und zwar mit dem Attribut "name". Der vergebene Name wird bei PHP-Scripts benötigt, um auf die Daten des Eingabefelds zuzugreifen.

Ergänzung 4/2008

5.2.5 Seite 4 Ferner sollten Sie bei einzeiligen Eingabefeldern immer die Anzeigelänge in Zeichen mit "size" sowie die interne Feldlänge in Zeichen mit "maxlength" bestimmen. Beide Angaben bedeuten die Anzahl der Zeichen ("size" = Größe; "maxlength" = maximal length = maximale Länge). Wenn die interne Feldlänge "maxlength" größer ist als die angezeigte Feldlänge "size" (wie im zweiten Eingabefeld des obigen Beispiels), dann wird bei längeren Eingaben automatisch gescrollt (im Beispiel also ab dem 31. eingegebenen Zeichen).

Mit dem zusätzlichen Attribut "value" können Sie einen Text bestimmen, mit dem das Feld vorbelegt wird ("value" = Wert).

#### 5.2.5.2.1.2 Radio Buttons

_		
	<input< td=""><td>type="radio"</td></input<>	type="radio"
		name="Zahlmethode"
		value="Mastercard">
		Mastercard
	<input< th=""><th>type="radio"</th></input<>	type="radio"
		name="Zahlmethode"
		value="Visa">
		Visa
	<input< th=""><th>type="radio"</th></input<>	type="radio"
		name="Zahlmethode"
		value="AmericanExpress">
		American Express

Radio-Buttons werden durch "<input type="radio">" definiert (input = Eingabe). Jeder Radio-Button sollte einen internen Bezeichnernamen erhalten – und zwar mit dem Attribut "name". Alle Radio-Buttons, die den gleichen Namen haben, gehören zu einer Gruppe, – das heißt: Von diesen Buttons kann der Anwender genau einen markieren. Der vergebene Name wird bei PHP-Scripts benötigt, um auf die Daten des Eingabefelds zugreifen zu können.

Mit dem Attribut "value" bestimmen Sie einen internen Bezeichnerwert für jeden Radio-Button ("value" = Wert). Wenn der Anwender das Formular abschickt, wird der Bezeichnerwert des markierten Buttons übertragen.

Vor oder hinter dem Tag "<input>" können Sie den Text notieren, der als Beschriftung der jeweiligen Option erscheint. Wenn Sie eine der Auswahlmöglichkeiten vorselektieren möchten, dann notieren Sie im Tag "<input>" des entsprechenden Radio-Button das alleinstehende Attribut "checked" – zum Beispiel:



**Ein ITP Handbuch** 

Dialoge erstellen

```
<input type="radio"
name="Typ"
value="Kassenpatient" checked>
```

Wenn Sie XHTML-Standard-konform arbeiten möchten, dann müssen Sie dieses Attribut in der Form von "checked="checked"" notieren. Mehr als eine Auswahlmöglichkeit dürfen Sie bei Radio-Buttons nicht vorselektieren.

#### 5.2.5.2.1.3 Submit- und Reset-Buttons

Mit einem "Submit"-Button wird letztendlich das Formular abgeschickt. Er wird wie folgt erstellt:

<input type="submit" />

Die Beschriftung des Button ist von Browser zu Browser unterschiedlich. In einigen Browsern steht "Abschicken" auf dem Knopf, in anderen wiederum "Submit".

Mit einem "Reset"-Button werden alle Formularelemente auf ihren Standardwert zurückgesetzt, genau wie Sie ihn beim Öffnen vorgefunden haben. Die Felder, die "checked="checked"" oder "selected="selected"" haben, werden auf diese Werte zurückgesetzt – beziehungsweise: sie werden selektiert. Bei allen anderen Feldern wird die Eingabe (zum Beispiel bei einzeiligen Eingabefelder) gelöscht. "Reset"-Buttons erstellt man wie folgt:

```
<input type="reset" />
```

Die Beschriftung des Button kann "Abbrechen" oder "Reset" sein.

IBM POWER Systems i Softwareentwicklung und -modernisierung

Die Beschriftung der Buttons kann man mit einem "Value"-Attribut ändern. Dies könnte dann so aussehen:

<input type="submit" value="Formular absenden" /> <input type="reset" value="Eingaben rückgängig machen" />

Wenn man in einem "Submit"-Button ein "name"-Attribut hinzufügt, so ist im PHP-Script die Beschriftung des "Submit"-Button vorhanden.







**Dialoge erstellen** 

5.2.5 Seite 6

```
<form action="file.php" method="post">
<select name="Member">
<option selected="selected">Bitte wählen</option>
<option value="2">Progman</option>
<option value="3">Erasor</option>
<option value="3">Erasor</option>
<option value="5">SuckOr</option>
<option value="6">Camper</option>
</select>
<input type="submit" name="formaction"
value="Löschen" />
<input type="submit" name="formaction"
value="Bearbeiten" />
<input type="reset" />
```

Im PHP-Script kann man anschließend den Wert auslesen und darauf entsprechend mit einer Abfrage reagieren, zum Beispiel wie hier: "Member löschen oder bearbeiten".

#### 5.2.5.2.2 Das Suchformular: suchesvklst00.php

Folgendes HTML-Formular wurde von mir erzeugt:

🕒 Kunden und Lieferantens 🚨	
Kunden und Lieferanten suchen	
Suchbegriff	
Kunden / Lieferanten	• •
	Suchen Initialisieren

#### Layout des Formulars

Das Formular wird durch das uns bereits bekannte "CSS" formatiert: "format.css".

Jetzt sehen Sie das zugrunde liegende HTML-Dokument:



**2** Kapite

5.2.5 Seite 7

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1" />
<LINK href="Styles/format.css" rel="stylesheet"
type="text/css">
<title>Kunden und Lieferantensuche</title>
</head>
<body>
<form id="sucheKL"
name="sucheKL"
method="post"
action="svklst00.php">
```

Im Formular-Header habe ich das bereits erstellte PHP-Script in das Attribut "action" eingetragen.

In das Formular wurde eine Tabelle eingebettet, die nicht sichtbar ist und die dem Zweck dient, den Dialog zu strukturieren.

```
Kunden und Lieferanten suchen
```

Kapitel <b>2</b>	<b>Web-Prog</b> Dialoge ers	<b>Jrammierung</b> tellen	
5.2.5		Überschriftenzeile:	
Seite 8			
		Suchber	griff
		<input< th=""><th>name="Such"</th></input<>	name="Such"
			type="text"
			value=""
			id="Such"
			size="60"
			<pre>maxlength="60" /&gt;</pre>

Suchbegriff-Eingabefeld:

```
Kunden / Lieferanten
 <input name="GPTyp"
          type="radio"
          value="KD"
          checked="checked"/>
    <input name="GPTyp"
          type="radio"
          value="LF"/>
```

Radio-Buttons zur Kunden bzw. Lieferantenauswahl:

```
 
      <input name="Suchen"
             type="submit"
             id="Suchen"
                               value="Suchen" />
```



Dialoge erstellen



Seite 9

"Submit"-Knopf, um die Eingabe an das verarbeitende Script zu übergeben:

```
<input name="Initialisieren"
type="reset"
id="Initialisieren"
value="Initialisieren"/>
```

"Reset"-Knopf, um die Eingabefelder zu initialisieren:

</form> </body> </html>

Wenn Sie jetzt das Script aufrufen, dann sollte der Aufruf von "svklst00.php" bereits erfolgreich ausgeführt worden sein.

Folgende URL trage ich in meinem Browser ein:

http://172.15.2.5.111.1:89/phponsystemi/suchesvklst00.php

Anschließend wird der Suchdialog angezeigt:

🕒 Kunden und Lieferantens	
Kunden und Lieferanten sucher	
Suchbegriff	
Kunden / Lieferanten	• •
	Suchen Initialisieren

Test des Formulars

Nach einem Mausklick auf den Knopf "Suchen" wird das im vorherigen Kapitel erstellte Script "svklst00.php" aufgerufen.

😳 Kunden und Lieferantenliste 🔽							
Identnr. Name	Land	PLZ	Ort	Strasse	Telefon	Kunde seit	
1000200 BT Deutschland GmbH	Deutschland	30853	Langenhagen	Grovestr. 16	0511/7262/0 e	2002-12-03	
1000500 Coty Deutschland GmbH Fort Malakoff Park	Deutschland	55116	Mainz	Rheinstr. 4e	06131/306920 e	2002-12-03	
1001200 ADP Employer Services GmbH	Deutschland	28357	Bremen	Im Hollergrund 3	0421/20710 e	2002-05-29	
1001500 Händle GmbH & Co. KG Maschinen und Anlagenbau	Deutschland	75417	Mühlacker	Industriestr. 47	07041/8911 e	2002-12-03	
1001700 F.W. Hempel & CO. Erze und Metalle	Deutschland	40211	Düsseldorf	Leopoldstraße 16	0211/168060 e	2002-05-29	
1001900 Unilog Integrata Training AG Computer und Management Trainings GmbH	Deutschland	81539	München	Schleifmühleweg 68	089/68089730	1997-11-06	
1002200 Lloyd Shoes GmbH & Co. KG	Deutschland	27232	Sulingen	Hans-Hermann-Meyer-Str. 1	04271/9400 e	2002-05-29	
1002300 Kaugummifabrik Bubble	Deutschland	63322	Rödermark	Paul-Ehrlich-Straße 34	12345	2002-12-03	
1002800 Pracht Spedition + Logistik	Deutschland	35708	Haiger	Hansastraße 10	02773/80/0 e	2002-05-29	
1003100 CIBA Vision Vertriebs GmbH	Deutschland	63868	Großwallstadt	Industriering 1	06022/240160 e	2002-12-02	
1003300 Beeck Feinkost GmbH & Co. KG	Deutschland	22045	Hamburg	Albert-Schweitzer-Ring 35	040/669430 e	2002-05-29	
ADDADD U.S. C. HUT H.S. P. J. C. H. JW	Deutenhland	20527	1 In male surger	Mandanaha 414 434	040/21044/0 -	2002 00 22	

Die Datentabelle



# 5.2.55.2.5.3Seite 10PHP: Parameterübergaben

Wie Sie bereits im vorherigen Kapitel erfahren haben, können Daten aus einem Formular über zwei verschiedene Methoden an einen verarbeitenden Prozess übergeben werden. Die erste Methode heißt "get", die andere "post". Im Folgenden betrachten wir zunächst die Methode "post".

Um überhaupt über die Methode "post" Daten an ein verarbeitendes Script zu übergeben, muss im Formular-Header "form" das Attribut "method=post" angegeben werden.

PHP übernimmt diese Daten in ein Array mit dem Namen "\$\_POST".

Um beispielsweise den Inhalt des Formularfelds "suche" zu verarbeiten, fragen wir im PHP-Script "svklst00.php" das Array-Element "\$\_POST['sucheKL']" ab.

Relevante Anweisung im PHP-Script "suchesvklst00.php":

```
<form id="sucheKL" name="sucheKL" method="post"
action="svklst00.php">
```

Relevante Anweisung im PHP-Script "svklst00.php":

```
$query = "select
                  klidnr,
                  klname,
                  (select
                          slbez
                   from
                           epsw0402.svslst00
                   where
                          slkey = 'LAND' and
                          slwert = kl.kllandkz) as klland,
                  klplz,
                  klort,
                  klstr,
                  kltel,
                  kldatn
          from
```





Seite 11

```
epsw0402.svklst00 kl
where
klidnr between 1000000 and 6999999 AND
upper(klname) like
upper('%".$_POST['such']."%')
order by
klidnr" ;
```

Testen wir nun unseren Dialog. Ich rufe dazu zuerst das Suchformular "suchsvklst00" auf:

```
http://172.15.2.5.111.1:89/phponsystemi/suchesvklst00.php
```

🕒 Kunden und Lieferantens 🚨 🗌	
Kunden und Lieferanten suchen	
Suchbegriff	
Kunden / Lieferanten	© C

#### Suchbegriff eingeben

Nachdem ich einen Suchbegriff eingegeben habe, klicke ich auf "Suchen". Daraufhin erscheint das Script "svklst00.php".



Gefundenen Daten anzeigen

Jetzt müssen die Radio-Buttons aktiviert werden. Im Folgenden können Sie den Quellcode für die Erzeugung der Radio-Buttons sehen.

```
<input name="GPTyp" type="radio" value="KD"
checked="checked"/>
<input name="GPTyp" type="radio" value="LF"/>
```

Diese beiden Zeilen bewirken Folgendes:

Wenn der Anwender auf den ersten Radio-Button klickt, so wird in das "POST"-Array der Wert "KD" gestellt. Klickt der Anwender auf den zweiten Radio-Button, so wird der Wert "LF" in das "POST"-Array gestellt.

Wir müssen also für das Script "svklst00.php" eine Entscheidung treffen:





**Dialoge erstellen** 

5.2.5 Seite 12

Wenn \$\_POST['GPTyp'] = "KD" dann
 Kunden abfragen (klidnr between 1.000.000 And
5.2.5.999.999
Sonst
Lieferanten abfragen ( klidnr >= 7.000.000

#### 5.2.5.3.1 Modifikation in "svklst00.php"

```
if ($_POST['GPTyp']== "KD") {
     $query = "select
                 klidnr,
                 klname,
                 (select slbez
                  from epsw0402.svs1st00
                  where slkey = 'LAND' and
                        slwert = kl.kllandkz) as klland,
                 klplz,
                 klort,
                 klstr,
                 kltel,
                 kldatn
              from
                 epsw0402.svklst00 kl
              where
                 klidnr between 1000000 and 6999999 AND
                 upper(klname) like
                 upper('%".$_POST['such']."%')
              order by
                 klidnr";
  }
```



```
else {
              $query = "select
              klidnr,
              klname,
              (select slbez from epsw0402.svslst00
               where slkey = 'LAND' and
                     slwert = kl.kllandkz) as klland,
              klplz,
              klort,
              klstr,
              kltel,
              kldatn
           from
              epsw0402.svklst00 kl
           where
              klidnr >= 7000000 AND
              upper(klname) like
              upper('%".$_POST['such']."%')
           order by
              klidnr";
}
```

Sicher könnten Sie diese SQL-Abfragen noch verfeinern, indem Sie weitere Suchfelder einfügen. Allerdings überlasse ich das Ihrer Fantasie.

## 5.2.5.3.2 Script SVKLSTOO.PHP - V1.4

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
          charset=iso-8859-1" />
```



Seite 13

```
Web-Programmierung
          Dialoge erstellen
5.2.5
                         <LINK href="Styles/format.css" rel="stylesheet"
Seite 14
                               type="text/css">
                         <title>Kunden und Lieferantenliste</title>
                       </head>
                       <body>
                       <?php
                        require("includes/localSettings.php");
                        $connectionstring = db2_connect($system,
                                                          $user,
                                                          $passwd);
                        // Verbindungsfehler abfangen und Script beenden
                        if (!$connectionstring) {
                         die("Status: ". db2_conn_error(). " " .
                              db2_conn_errormsg());
                        }
                        if ($_POST['GPTyp']== "KD") {
                          $query = "select
                                      klidnr,
                                      klname,
                                       (select slbez from epsw0402.svslst00
                                       where slkey = 'LAND' and
                                       slwert = kl.kllandkz) as klland,
                                      klplz,
                                      klort,
                                      klstr,
                                      kltel,
```



		Ka
ì	5 2	5

pitel

	kldatn	5.2.5
	from	Seite 15
	epsw0402.svklst00 kl	
	where	
	klidnr between 1000000 and 6999999 AND	
	upper(klname) like	
	upper('%".\$_POST['such']."%')	
	order by	
	klidnr";	
}		
else {		
	\$query = "select	
	klidnr,	
	klname,	
	(select slbez from epsw0402.svslst00	
	where slkey = 'LAND' and	
	slwert = kl.kllandkz) as klland,	
	klplz,	
	klort,	
	klstr,	
	kltel,	
	kldatn	
	from	
	epsw0402.svklst00 kl	
	where	
	klidnr >= 7000000 AND	
	upper(klname) like	
	upper('%".\$_POST['such']."%')	
	order by	
	klidnr";	





\$klstr

\$kltel \$kldatn



= db2\_result(\$queryexe, 'KLSTR'); = db2\_result(\$queryexe, 'KLTEL');

= db2\_result(\$queryexe, 'KLDATN');



#### 5.2.5.4 Ein anderer Lösungsansatz

Eigentlich könnten Sie mit dem bisher erstellten Dialog zufrieden sein. Dennoch: Sie finden einerseits in vielen Web-Seiten Suchdialoge, die nach dem Muster des von uns erstellten Dialogs funktionieren. Aber ist es andererseits wirklich optimal, den Dialog in zwei unterschiedlichen Seiten unterzubringen? Diese Dialogabfolge verlangt vom Anwender (wenn eine erste Selektion durchgeführt und das Ergebnis angezeigt wurde), dass für eine neue Selektion zuerst der Zurück-Knopf im Browser gedrückt werden muss. Erst dann kann eine neue Abfrage gestartet werden.

Nach ergonomischen Gesichtspunkten ist das ein suboptimaler Zustand!

Lassen Sie uns dennoch beide Dialoge in einem Dokument zusammenlegen. Dazu können wir jedoch nicht einfach beide PHP-Scripte zusammenkopieren.



**Dialoge erstellen** 

# 5.2.5 Seite 18

Warum eigentlich nicht? Die Antwort auf diese Frage liegt in der Abarbeitung eines PHP-Scripts: Ein PHP-Script wird immer vollständig von der ersten bis zur letzten Anweisung ausgeführt.

Wenn wir uns jetzt vorstellen, dass wir beide Scripts zusammengeführt hätten, dann würde zuerst das Suchformular angezeigt werden. Anschließend müsste jedoch sofort die Abfragelogik ausgeführt werden, die allerdings nicht so einfach funktioniert, da wir zuerst das Suchformular angezeigt bekommen. Anschließend muss sich das Script selbst aufrufen; beim zweiten Durchlauf muss das Formular und die Abfragelogik ausgeführt sowie das Abfrageergebnis angezeigt werden.

Um das zu erreichen, müssen wir eine Übersprungslogik entwerfen, die dafür aufkommt, dass im ersten Dokument-Durchlauf das Formular angezeigt wird. Im zweiten werden dann Formular und Ergebnis der Abfrage angezeigt.

Wie erreichen wir das? Wie können wir unterscheiden, ob es sich beim Dokumentaufruf um den Erst- oder den Zweitdurchlauf handelt?

Im PHP ist für solchen Dialogtyp eine Technik im Einsatz, die sich an den Zustand einer Variablen orientiert. Im PHP-Umfeld können Variablen zwei Zustände haben:

- gesetzt
- nicht gesetzt

Eine Variable gilt als nicht gesetzt, wenn ihr kein Wert zugewiesen wurde. Eine Variable gilt als gesetzt, wenn ihr einmal ein Wert zugewiesen wurde.

Welche Variablen sind in unserem Script "svklst00.php" anfänglich gesetzt worden? Hier die Antwort:

```
$system = "ITPSERVER";
$user = "WWWUSER";
$passwd = "INTERNET";
```

Diese Variablen sind gesetzt, da ihnen ein Wert zugewiesen wurde. Das heißt: Wir können sie nicht für unsere Steuerlogik gebrauchen.

Dieser Sachverhalt führt uns zur nächsten Frage. Welche Variablen gelten in unserem neuen Script als nicht gesetzt? Die Antwort: Alle Variablen, die Formularfelder repräsentieren.

```
$_POST['such']
$_POST['GPTyp']
```





Seite 19

Dialoge erstellen

Wenn das Formular erstmalig angezeigt wird, dann ist diesen Feldern noch kein Wert zugewiesen worden - das heißt, dass sie nicht gesetzt sind. Erst dann, wenn wir im Formular eine Eingabe gemacht haben, gilt diese Eingabe als gesetzt.

Die Ablauflogik könnte unser neues Script, das wir "svklst01.php" nennen, wie folgt beschreiben:

```
Wenn $_POST['such'] nicht gesetzt dann
       Formular anzeigen.
Sonst
       Formular anzeigen
       Abfrage ausführen
       Tabelle anzeigen
```

# 5.2.5.5 Script SVKLST01.PHP - V1.0

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html;
     charset=iso-8859-1" />
```

<LINK href="Styles/format.css" rel="stylesheet" type="text/ css">

<title>Kunden und Lieferantenliste</title> </head>

<body>

<?php

Ergänzung 4/2008

require("includes/localSettings.php");

Im folgenden Code wird das Formular ausgegeben:



Kapitel

```
Web-Programmierung
```

Dialoge erstellen

5.2.5 Seite 20

```
print("<form id=\"sucheKL\"</pre>
           name = "sucheKL"
           method=\"post\"
           action=\"svklst01.php\">");
 print("");
  print("");
    print("Kunden und Lieferanten suchen");
    print(" ");
  print("");
  print("");
    print("Suchbegriff");
    print("<input name=\"such\"</pre>
                    type=\"text\"
                    value=\"\"
                    id = \"such \"
                    size=\"60\"
                    maxlength=\"60\"/>");
  print("");
  print("");
    print("Kunden / Lieferanten");
    print("");
        print("<input name=\"GPTyp\"</pre>
                    type=\"radio\"
                    value=\"KD\"
                    checked=\"checked\"/>");
        print("<input name=\"GPTyp\"</pre>
                    type=\"radio\"
                    value=\"LF\"/>");
    print("");
  print("");
  print("");
    print(" ");
```



5.2.5 Seite 21

```
print("");
    print("<input name=\"Suchen\"
        type=\"submit\"
        id=\"Suchen\"
        value=\"Suchen\" />");
    print("<input name=\"Initialisieren\"
        type=\"reset\"
        id=\"Initialisieren\"
        value=\"Initialisieren\"
        value=\"Initialisieren\";
    print("</td>
```

Anschließend finden Sie die Abfrage zum Überspringen der Datenbank-Abfragelogik sowie der tabellarischen Anzeige der Daten.



| <b>L</b>          | Web-Prog    | <b>Jrammierung</b> |  |
|-------------------|-------------|--------------------|--|
| Ка                | Dialoge ers | tellen             |  |
| 5.2.5<br>Seite 22 |             |                    | (select slbez from epsw0402.svslst00   |
|                   |             |                    | where slkey = 'LAND' and               |
|                   |             |                    | slwert = kl.kllandkz) as klland,       |
|                   |             |                    | klplz,                                 |
|                   |             |                    | klort,                                 |
|                   |             |                    | klstr,                                 |
|                   |             |                    | kltel,                                 |
|                   |             |                    | kldatn                                 |
|                   |             |                    | from                                   |
|                   |             |                    | epsw0402.svklst00 kl                   |
|                   |             |                    | where                                  |
|                   |             |                    | klidnr between 1000000 and 6999999 AND |
|                   |             |                    | upper(klname) like                     |
|                   |             |                    | upper('%".\$_POST['such']."%')         |
|                   |             |                    | order by                               |
|                   |             |                    | klidnr";                               |
|                   |             | }                  |  |
|                   |             | else {             |  |
|                   |             |                    | <pre>\$query = "select</pre>           |
|                   |             |                    | klidnr,                                |
|                   |             |                    | klname,                                |
|                   |             |                    | (select slbez from epsw0402.svslst00   |
|                   |             |                    | where slkey = 'LAND' and               |
|                   |             |                    | slwert = kl.kllandkz) as klland,       |
|                   |             |                    | klplz,                                 |
|                   |             |                    | klort,                                 |
|                   |             |                    | klstr,                                 |
|                   |             |                    | kltel,                                 |
|                   |             |                    | kldatn                                 |
|                   |             |                    | from                                   |



Seite 23

```
epsw0402.svklst00 kl
           where
              klidnr >= 7000000 AND
              upper(klname) like
              upper('%".$_POST['such']."%')
           order by
              klidnr";
}
 $queryexe = db2_exec($connectionstring, $query);
if (!$queryexe) {
 die("Status: ". db2_stmt_error(). " " .
                 db2_stmt_errormsg());
}
 print("<TABLE width=\"95%\" border=\"0\">");
   print("<TBODY>");
     print("<TR>");
       print("<TH>Identnr.</TH>");
       print("<TH>Name</TH>");
                 print("<TH>Land</TH>");
                 print("<TH>PLZ</TH>");
                 print("<TH>Ort</TH>");
                 print("<TH>Strasse</TH>");
                 print("<TH>Telefon</TH>");
                 print("<TH>Kunde seit</TH>");
                 print("</TR>");
 while(db2_fetch_row($queryexe))
 {
       $klidnr
                 = db2_result($queryexe, 'KLIDNR');
```



Ergänzung 4/2008

| e   |  |
|-----|--|
| pit |  |
| Ka  |  |

Dialoge erstellen

| 5.2.5   |   |
|---------|---|
| Seite 2 | 4 |

|       | \$klname  | <pre>= db2_result(\$queryexe,</pre> | 'KLNAME'); |
|-------|---|-------------------------------------|------------|
|       | \$k11and  | <pre>= db2_result(\$queryexe,</pre> | 'KLLAND'); |
|       | \$klplz   | <pre>= db2_result(\$queryexe,</pre> | 'KLPLZ');  |
|       | \$klort   | <pre>= db2_result(\$queryexe,</pre> | 'KLORT');  |
|       | \$klstr   | <pre>= db2_result(\$queryexe,</pre> | 'KLSTR');  |
|       | \$kltel   | <pre>= db2_result(\$queryexe,</pre> | 'KLTEL');  |
|       | \$kldatn  | <pre>= db2_result(\$queryexe,</pre> | 'KLDATN'); |
|       |   |                                     |            |
|       |   |                                     |            |
|       | print(" <t< td=""><td>R&gt;");</td><td></td></t<> | R>");                               |            |
|       | print(" <t< td=""><td>D&gt;\$klidnr</td>");</t<>  | D>\$klidnr                          |            |
|       | print(" <t< td=""><td>D&gt;\$k1name</td>");</t<>  | D>\$k1name                          |            |
|       | print(" <t< td=""><td>D&gt;\$klland</td>");</t<>  | D>\$klland                          |            |
|       | print(" <t< td=""><td>D&gt;\$klplz </td>");</t<>  | D>\$klplz                           |            |
|       | print(" <t< td=""><td>D&gt;\$klort </td>");</t<>  | D>\$klort                           |            |
|       | print(" <t< td=""><td>D&gt;\$klstr </td>");</t<>  | D>\$klstr                           |            |
|       | print(" <t< td=""><td>D&gt;\$kltel </td>");</t<>  | D>\$kltel                           |            |
|       | print(" <t< td=""><td>D&gt;\$kldatn</td>");</t<>  | D>\$kldatn                          |            |
|       | print(" </td <td>TR&gt;");</td> <td></td>         | TR>");                              |            |
| }     |   |                                     |            |
| pri   | nt(" <td>Y&gt;");</td> <td></td>                  | Y>");                               |            |
| print | ("  |                                     |            |

 "); |  ||  |  |  |  |
|  |  |  |  |
| db2\_c | lose(\$conn | ectionstring); |  |
|  |  |  |  |
| } |  |  |  |
| ?> |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |




Seite 25

#### 5.2.5.6 Optimierungen 1

Im Folgenden sehen wir uns den Dialog des Scripts "svklst01.php" an. Dabei werden wir uns fragen, was sich an dem Script verbessern lässt.

unden und Lieferanten s	suchen	
uchbegriff	ds	
unden / Lieferanten	e c	
	Suchen Initialisieren	

Erstaufruf des Scripts "svklst01.php"

Wir geben einen Suchbegriff ein und klicke auf "Suchen".

Kunden u	und Lieferanten suchen							
Suchbegri	ff							
Kunden / I	Lieferanten	• •						
		Suchen Initialisieren IIII						
Identnr.	Name		Land	PLZ	Ort	Strasse	Telefon	Kunde seit
1014500	Kraft Foods Deutschland	GmbH & Co. KG	Deutschland	28199	Bremen	Langemarckstr. 4-20	0421/59901 e	2002-12-03
1027400	Harrods Knightsbridge L	imited	Groß Britannien	SWIX 7X	London	Knightsbridge	0044/1717301234	2002-05-29
1042000	Berendsohn AG		Deutschland	22559	Hamburg	Alte Sülldorfer Landstr. 400	040/81909106 e	2002-05-29
1064000	064000 JD Edwards Deutschland GmbH		Deutschland	64546	Mörfelden/Frankfurt	Waldeckerstrasse 11	06103/7260 e	2002-12-11
1089900	Wort+Bild Verlag Konra	dshöhe GmbH & Co.	Deutschland	82065	Baierbrunn	Konradshöhe	089/744330 e	2002-12-12
1142600	ADS Informatik Hard- u	. Software-Beratung	Deutschland	66693	Mettlach	Goetheweg 8	06865/93300 e	2002-12-12
1202600 Smith + Nephew Richards GmbH Medizintechnik			Deutschland	78532	Tuttlingen	Alemannenstr. 14	07462/2080 e	2003-02-11
1205800 DIFA Deutsche Immobilien Fonds AG			Deutschland	20354	Hamburg	Valentinskamp 20	040/349190 e	2002-05-29
1224500	1224500 HUDSON Intern. Vertr. GmbH		Deutschland	70565	Stuttgart	Am Wallgraben 45	0711/78150	2002-05-29
1231200	Holert Kalksandsteinwer	ke GmbH & Co.	Deutschland	24568	Kaltenkirchen	Barmstedter Str. 14	04191/7050 e	2002-05-29
1257600	1257600 Fonds f. Umweltstudien e.V.		Deutschland	53175	Bonn	Godesberger Allee 108 - 112	0228/2692/0	2002-05-29
1272200	IDS Inter Data		Deutschland	63477	Maintal	Am Kreuzstein 82-84	06109/76680	2002-05-29
1311600	1311600 TDS Tele Columbus Daten und Service GmbH		Deutschland	30519	Hannover	Peiner Str. 8	0511/98765 e	2002-05-29
1317100	317100 Virgin Records GmbH		Deutschland	80803	München	Herzogstr. 64	089/381950	2002-05-29
1317200	Gutehoffnungshütte Rad	Isatz GmbH	Deutschland	46145	Oberhausen	Steinbrinkstr. 170	0208/7400/0 e	2002-05-29
1335000	335000 DSM Computer		Deutschland	57299	Burbach	Forstweg 14	02736/3176	2003-04-22
1352900	52900 IDS GmbH		Deutschland	69517	Gorxheimertal	Am Wetzelsberg 2	06201/292091 e	2003-04-15
1353200	Berendsen PMC GmbH S	teinebronn Industriebedarf Geschäftsbereich	Deutschland	70499	Stuttgart	Kranstr. 9	0711/8361/0	2002-05-29
1377900	Verbandsgemeindeverw	altung EDV-Organisation	Deutschland	57610	Altenkirchen	Rathausstraße 13	02681/850	2002-05-29
1380100	Harley-Davidson GmbH		Deutschland	64546	Mörfelden-Walldorf	Starkenburgstr. 12	06105/2840	2002-05-29
1385000	Feige, Siegfried LKW-Ins	itandsetzung	Deutschland	02708	Löbau	Weißenberger Str. 45	03585/862537 e	2002-10-09
1434300	Mundschenk Druck- und	Verlags- gesellschaft mbH	Deutschland	29614	Soltau	Harburger Str. 63	05191/808/0 e	2002-02-18
1464700	Unilever Best Foods Deu	tschland GmbH	Deutschland	20355	Hamburg	Dammtorwall 15	040/3493/0 e	2002-02-06
1483200	HDS Rechenzentrum Gr	nbH & Co. KG	Deutschland	49577	Ankum	Am Lordsee 1	05462/765/0 e	2002-08-05
1498400	Randstad Organisation f	ür Zeitarbeit GmbH	Deutschland	65760	Eschborn	Ginheimerstr. 5	06196/408/0 e	2003-01-09
2032801	Deutsche Immobilien Fo	nds AG DIFA	Deutschland	20355	Hamburg	Caffamacherreihe 8	040/34919-0 e	2003-01-14
2045201	WestInvest Gesellschaft	für Investmentfonds mbH	Deutschland	40476	Düsseldorf	Hans-Böckler-Str. 33	0211/51798-0 e	2002-06-05
2054401	Fonds für Umweltstudie	n e.V.	Deutschland	53175	Bonn	Godesbergerallee 108-112	0228/2692216 e	2003-06-02
2054701	EDS Informationstechno	logie & Service GmbH	Deutschland	53229	Bonn	Kautexstr. 52	0228/488-0 e	2002-11-06

2054701 EDS Informationstechnologie & Service GmbH
Anzeige des Gesamtdialogs

Die Anzeige des gesamten Dialogs funktioniert. Störend ist, dass im Suchformular die Suchkriterien nicht weiter angezeigt werden.





5.2.5 Seite 26 Dieses Problem lässt sich leicht durch die Modifikation des INPUT-Elements beheben:

Ich habe hier einfach das Element "\$\_POST" in das Attribut "value" des Formularfelds eingetragen, so dass das Eingabefeld mit seinem eigenen Inhalt initialisiert wird.

Ähnlich verfahre ich mit dem Radio-Button "GPTyp", der immer den aktuell markierten Button anzeigen soll.

```
print("<input name=\"GPTyp\"
    type=\"radio\"
    value=\"KD\"
    checked=\"checked\"/>");
    if ($_POST['GPTyp'] == "LF"){
        print("<input name=\"GPTyp\"
        type=\"radio\"
        value=\"LF\"
        checked=\"checked\"/>");
    }
    else {
    print("<input name=\"GPTyp\"
        type=\"radio\"
        value=\"LF\"/>");
    }
}
```

In diesem Beispiel habe ich eine Art ,Flip-Flop-Logik' angewendet, die dafür sorgt, dass er, wenn der Lieferanten-Button angeklickt wird, das Attribut "checked = "checked"" erhält.

Was jetzt aber leider nicht mehr funktioniert, ist der Knopf "Initialisieren". Aus diesem Grund werden wir uns ein wenig mit Javascript beschäftigen müssen.





Seite 27

#### 5.2.5.7 **Exkurs Javascript**

In diesem Exkurs geht es nicht darum, Ihnen die Javascript-Syntax grundsätzlich zu erklären. Vielmehr steht im Vordergrund, wie man durch Javascript (im Folgenden JS) die Anwendungsdialoge besser bedienbar machen kann.

Dazu werden wir folgende Bereiche näher betrachten:

- Initialisieren von Dialogen
- Statische Validierung von Dialogen
- Einheitliche Nachrichtendarstellung

Sie werden sich möglicherweise fragen, warum das nicht alles in PHP zu erledigen ist. Grundsätzlich lassen sich die oben beschriebenen Aufgabenbereiche durchaus in PHP realisieren, doch für die Ergonomie unserer Anwendungsdialoge stellt Javascript eine Reihe erheblicher Vorteile dar.

Zunächst ist erst einmal zu klären, was Javascript eigentlich ist. Javascript ist kein direkter Bestandteil von HTML, sondern eine eigene Programmiersprache. Diese Sprache wurde eigens zu dem Zweck geschaffen, HTML-Autoren ein Werkzeug in die Hand zu geben, um Web-Seiten zu optimieren.

Javascripts werden entweder direkt in der HTML-Datei oder in separaten Dateien notiert. Sie werden zur Laufzeit vom Web-Browser interpretiert - im Gegensatz zu PHP, das vom Web-Server interpretiert wird. Dazu besitzen moderne Web-Browser entsprechende Interpreter-Software.

Javascript ist im Vergleich zu anderen eine einfache Sprache, bei der viele Aufgabenbereiche einer 'großen' Programmiersprache fehlen – zum Beispiel Elemente wie "Arbeitsspeicherverwaltung" oder "Dateioperationen". Außerdem setzt Javascript auf einer bestimmten Umgebung auf – nämlich auf einer anzuzeigenden oder angezeigten Web-Seite.

Javascript läuft in einer sogenannten "Sandbox". Das ist eine Art Sicherheitskäfig, in dem die Programmiersprache eingesperrt ist. Sie wird dabei um typische Möglichkeiten anderer Programmiersprachen beschnitten, vor allem um die Möglichkeit, beliebig Daten aus Dateien zu lesen und in Dateien schreiben zu können. So soll verhindert werden, dass Javascript-Programmierer auf den Rechnern von Anwendern, die eine Javascript-unterstützte Web-Seite aufrufen, Unfug betreiben können. Wer Javascript einsetzt, sollte die Möglichkeiten der Sprache so verwenden, dass der Anwender einen Mehrwert davon hat.

Bevor Sie sich also in die Tiefen von Javascript stürzen, sollten Sie sich bewusst machen, dass Javascript wirklich nur als Ergänzung zu normalem HTML gedacht ist, nicht als dessen Ersatz. Es ist durchaus möglich, Web-Seiten ausschließlich in Javascript zu programmieren. Dann kann man kaum noch davon reden, dass Javascript in HTML eingebettet ist, sondern eher davon, dass der HTML-Code der Seite mit Hilfe von Javascript dynamisch erzeugt wurde.



## 5.2.5 5.2.5.7.1 Seite 28 Einbettung

Neben der direkten Einbettung von Javascript-Syntaxen in die HTML Syntax, kann Javascript auch extern abgelegt und aus HTML heraus referenziert werden. Nur die letztgenannte Variante werden wir verwenden.

<html></html>
<head></head>
<title>JavaScript-Test</title>
<pre><script src="js/Formhandling.js" type="text/javascript"></script></pre>

Sie sehen im oben dargestellten HTML-Codefragment, dass die Referenz zur Javascript-Datei "Formhandling.js" in das Tag "script" eingebettet wird. Das "script"-Tag ist wiederum eingebettet in das "head"-tag.

Vergleichen Sie die Javascript- mit der CCS-Referenzierung – und Sie erkennen Parallelen.

## 5.2.5.8 Funktionen

Eine Javascript-Datei wie "Formhandling.js" kann dann aus 1 bis n Javascript-Funktionen bestehen. Eine Javascript-Funktion ist vergleichbar mit einer RPG-Subroutine beziehungsweise Prozedur, die dann aus dem HTML-Dokument heraus aufgerufen wird.

Beginnen wir mit folgendem Beispiel: Die Aufgabe besteht darin, eine Message-Box anzuzeigen, die einen Text beinhaltet, der auf eine Fehlerkondition aufmerksam macht. Diese Funktion soll "ZeigeMessageBox" heißen.

```
function ZeigeMessageBox(Text){
```

Eine Funktion beginnt mit dem Schlüsselwort "function" – gefolgt vom Funktionsnamen und den zu übergebenen Parametern. In diesem Fall die Variable "Text". Damit ist der Text deklariert.

Der Funktionsrumpf wird durch die Zeichen "{ .... Anweisungen...}" gekennzeichnet.

}

}



In der Funktion soll eine Fehlernachrichtenbox angezeigt werden. Dafür verwenden wir die JS-Builtin-Funktion "alert()".

Jetzt sieht unsere Funktion wie folgt aus:

Damit ist unsere Funktion fertig.

Wie wird jetzt die Funktion "ZeigeMessageBox" aufgerufen? Dazu muss ein HTML-Formular folgendermaßen kodiert werden.

```
<body>
<form name="Formular" action="">
<input type="text" name="Eingabe" size="10">
<input type="button" value="Messagebox anzeigen" onclick="Zei-
geMessageBox('Fehler!')">
</form>
</body>
```

Beachten Sie das markierte Element "input". HTML-Input-Elemente unterstützen eine Ereignisabfrage. Klickt der Anwender auf den Button "Messagebox anzeigen", so wird das Ereignis "onclick" ausgelöst, dass dann die Funktion "ZeigeMessageBox" mit dem Parameterwert "'Fehler!'" auf aufruft. Die Zeichenfolge "'Fehler!'" wird an den Parameter "Text" übergeben – und "Text" wird dann mit "alert(Text)" ausgegeben.

Soweit erst einmal. Jetzt möchten wir jedoch nicht nur statische Meldungen ausgeben, sondern eingegebene Feldwerte in einer JS-Funktion verarbeiten.

Dazu müssen wir das sogenannte HTML Document Object Model (DOM) kennenlernen.

Sehen Sie sich einmal folgendes HTML-Dokument an:

Ergänzung 4/2008



Dialoge erstellen

5.2.5 Seite 30 <html> <html> <body> <form name="Form1" action=""> <input type="text" name="Eingabe" size="10"> </form> </html>

Ich habe hier alles weggelassen, was für das Verständnis des DOM uninteressant ist.

DOM geht davon aus, dass sich jedes relevante syntaktische Element eines HTML-Dokuments in einer Verzeichnisstruktur abbilden lässt.

Die höchste Hierarchieebene eines HTML Dokuments – "root" – wird bezeichnet durch "document".

document

Innerhalb von "document" finden wir ein HTML-Formular mit dem Namen "Form1".

document.Form1.

Innerhalb des Formulars findet sich ein Eingabefeld mit dem Namen "Eingabe".

document.Form1.Eingabe

Wenn wir den Wert, den ein Anwender in das Feld "Eingabe" gemacht hat, ansprechen wollen, dann sprechen wir seinen Wert an: "value".

document.Form1.Eingabe.value

Wenn wir also beabsichtigen, den Wert von "Eingabe"an die Funktion "Zeige-MessageBox" zu übergeben, dann müsste unser Funktionsaufruf wie folgt aussehen:





```
<html>
<body>
<form name="Form1" action="">
<input type="text" name="Eingabe" size="10">
<input type="button" value="Messagebox anzeigen" onclick="Zei-
geMessageBox(document.Form1.Eingabe.value)">
</form>
</html>
```

Eine weitere Aufgabe besteht darin, eine JS-Funktion zu realisieren, die prüft, ob in einem Eingabeformular bestimmte Felder gefüllt wurden. Die Besonderheit dieser Funktion ist, dass nicht ein einzelnes Feld an die Funktion übergeben wird, sondern alle Formularfelder. Zu einer weiteren Besonderheit gehört es, dass diese Funktion dann aufgerufen wird, wenn der Submit-Button gedrückt wurde. Eine dritte und letzte Besonderheit besteht darin, dass die Funktion, wenn sie einen Fehler entdeckt, die Methode "POST" nicht ausführt. Wenn die Funktion keinen Fehler entdeckt, wird die Methode "POST" ausgeführt; die in das Formular eingegebenen Werte können dann weiter verarbeitet werden.

Aufruf der Funktion aus dem HTML-Formular:

```
<html>
<body>
<form name="Form1"
action=""
method="post"
onsubmit=\"javascript:return validate(this)>\">
<input type="text" name="Eingabe" size="10">
<input type="text" name="Eingabe" size="10">
</form>
</form>
```

Beachten Sie den markierten Text. Wenn der Anwender auf den "Submit"-Button klickt, dann wird die Funktion "validate()" aufgerufen. Besonderheit: Das Schlüsselwort "return" bewirkt, dass von der Funktion "validate()" ein Rückkehrwert erwartet wird, der steuert, ob "submit" ausgeführt (kein Eingabefehler) oder zurückgewiesen wird (Eingabefehler) wird.

Das Schlüsselwort "javascript:" ist optional.

Der Parameterwert "this" in der Funktion "validate(this)" ist ein Sonderwert, der besagt, dass das ganze Formular "Form1" an die Funktion "validate()" als

Kapitel	Web-Programmierung Dialoge erstellen
5.2.5	Parameterwert übergeben wird und damit auch die Werte aller Eingabefelder.
Seite 32	Wie muss nun unsere Funktion "validate()" kodiert werden?

```
function validate(form){
```

An die Funktion "validate()" wird die Variable "form" übergeben. Damit ist "form" deklariert; "form" beinhaltet jetzt die Referenz auf das komplette Formular.

Um in der Funktion "validate(form)" das Feld "Eingabe" und den darin enthaltenen Wert anzusprechen, muss ich mit folgender Syntax arbeiten.

Form.Eingabe.value

Die Prüfbedingung sollte dann wie folgt aussehen:

```
if(Form.EINGABE.value == ""){
    alert("Ein Auswahlbegriff muss eingegeben werden");
    form.Eingabe.focus();
    return false;
}
```

Für den Fall, dass das Eingabefeld leer ist, wird die "Messagebox" mit "alert(…)" ausgegeben.

Mit der Funktion "focus()" wird der Kursor in das Feld "Eingabe" gestellt. Dann wird der Wert "false" an das HTML-Formular zurückgegeben und damit die Methode "POST" nicht ausgeführt.

return true;

}

Falls das Feld "Eingabe" einen Wert enthält, so gibt die Funktion "validate()" "true" zurück. Die Methode "POST" wird ausgeführt und die eingegebenen Daten können weiter verarbeitet werden.





Seite 33

#### 5.2.5.9 Optimierungen 2

Im Folgenden werde ich nun die Javascript-Funktion "validate()" in unser PHP-Script "svklst01.php" integrieren.

Die Funktion "validate()" soll hier prüfen, ob der Anwender in das Feld "such" einen Wert eingegeben hat, Wenn nicht, so soll die Nachricht ""Bitte Wert eingeben"" als "Messagebox" ausgegeben werden. Wird die Nachricht mit "OK" bestätigt, so soll der Cursor im Eingabefeld stehen.

Die Arbeit beginnt damit, dass ich in meinem Projekt einen Unterordner "/js" anlege; in "/js" erzeuge ich die Datei "formhandling.js".

In "formhandling.js" kodiere ich folgende Funktion:

function validate(form){

```
if(form.such.value == "") {
    alert("Bitte Wert eingeben.");
    form.such.focus();
        return false;
    }
    Return true;
}
```

In das Script "svklst01.php" füge ich die JS-Script-Referenz ein:

```
<head>
    <meta http-equiv="Content-Type" content="text/html;
         charset=iso-8859-1" />
         <LINK href="Styles/format.css" rel="stylesheet"
            type="text/css">
            <script src="js/formhandling.js"
            type="text/javascript"></script>
            <title>Kunden und Lieferantenliste</title>
            </head>
```



pitel 2	Web-Prog	rammierung
Ka	ellen	
5.2.5	Ι	Der Aufruf von "validate()" sieht wie folgt aus:
Seite 34		print(" <form id='\"sucheKL\"&lt;/td'></form>
		$name = \"sucheKL \"$
		<pre>method=\"post\"</pre>

action=\"svklst01.php\"

Der Test zeigt folgendes Ergebnis:

	Die Seite mit der Adresse http://eposi501.eposhbi
Suchen Initialisieren	Bitte Wert eingeben.
	ОК
	© C Suchen Initialisieren

onsubmit=\"javascript:return validate(this)\">");

*Eingabefehler!* 

#### 5.2.5.10 Optimierungen 3

In der nächsten Phase der Optimierung aktiviere ich die Taste "Initialisieren".

Ziel ist es, dass das Feld "such" geleert wird, wenn der Anwender auf "Initialisieren" klickt.

Ich schreibe die JS-Funktion "initialize()".

```
function initialize(form){
    form.such.value = "";
    form.such.focus();
    return false;
}
```


Der Aufruf der Funktion wird an den Knopf "Initialisieren" gebunden.

```
print("<input name=\"Initialisieren\"
    type=\"button\"
    id=\"Initialisieren\"
    value=\"Initialisieren\"
    onclick =\"javascript:return
        initialize(sucheKL)\"/>");
```

## 5.2.5.11 Optimierungen 4

Wenn Sie Script "svklst01.php" aufrufen, werden Sie feststellen, dass Sie zuerst einmal in das Feld "Suchbegriff" hineinklicken müssen, damit der Cursor in das Feld gestellt wird.

Das ist für einen Anwender dieses Dialogs sicher keine schöne Tätigkeit. Aber auch bei der Lösung dieses Problems hilft uns JS.

Sie wissen ja bereits, dass mit der Funktion "focus()" des JS der Cursor in ein Feld gestellt werden kann. Um den Cursor aber auch gezielt in das Feld "such" zu stellen, schreibe ich folgende JS-Funktion:

function pageHandler() {

```
document.sucheKL.such.focus();
```

return true;

}

Diese Funktion soll immer dann aufgerufen werden, wenn mein Script "svklst01. php" geladen wird.

Zu diesem Zweck ändere ich das HTML-Tag "body" wie folgt:

```
print("<body onload=\"javascript:return pageHandler();\">");
```

Auch das Tag "body" kennt Ereignisse; für den genannten Zweck bietet sich das Ereignis "onload" an. Immer dann, wenn "body" des HTML-Dokuments geladen wird, wird die Funktion "pageHandler()" aufgerufen. Sie bewirkt, dass der Cursor in das Feld "such" gestellt wird.



Dialoge erstellen

5.2.5 Seite 36

#### 5.2.5.12 Das Script SVKLST01.PHP - V1.1

Nach diesen Optimierungen sieht das Script "svklst01.php" wie folgt aus:

```
<html>
 <head>
   <meta http-equiv="Content-Type" content="text/html;
         charset=iso-8859-1" />
   <LINK href="Styles/format.css" rel="stylesheet"
         type="text/css">
   <script src="js/formhandling.js"</pre>
           type="text/javascript"></script>
   <title>Kunden und Lieferantenliste</title>
 </head>
 <?php
  require("includes/localSettings.php");
 print("<body onload=\"javascript:return</pre>
                           pageHandler();\">");
  print("<form id=\"sucheKL\" name=\"sucheKL\"</pre>
               method=\"post\" action=\"svklst01.php\"
               onsubmit=\"javascript:return
                                 validate(this)\"/>");
    print("");
     print("");
       print("Kunden und Lieferanten suchen");
       print(" ");
     print("");
```



5.2.5 Seite 37

```
print("");
  print("Suchbegriff");
  print("<input name=\"such\"</pre>
                   type=\"text\"
                   value=\"".$_POST['such']."\"
                   id = \"such \"
                   size = \"60 \"
                   maxlength=\"60\" />");
print("");
print("");
  print("Kunden / Lieferanten");
 print("");
     print("<input name=\"GPTyp\"</pre>
                   type=\"radio\"
                   value=\"KD\"
                   checked=\"checked\"/>");
     if ($_POST['GPTyp'] == "LF"){
                   print("<input name=\"GPTyp\"</pre>
                   type=\"radio\"
                   value=\"LF\"
                   checked=\"checked\"/>");
     }
     else {
                   print("<input name=\"GPTyp\"</pre>
                   type=\"radio\"
                   value=\"LF\"/>");
    }
  print("");
print("");
print("");
  print(" ");
  print("");
```



<b>Pitel</b>	Web-Programmierung						
Ка	Dialoge erste	ellen					
5.2.5 Seite 38		print(" <input< th=""><th><pre>name=\"Suchen\" type=\"submit\" id=\"Suchen\" value=\"Suchen\" /&gt;");</pre></th></input<>	<pre>name=\"Suchen\" type=\"submit\" id=\"Suchen\" value=\"Suchen\" /&gt;");</pre>				
		print(" <input< th=""><th><pre>name=\"Initialisieren\" type=\"reset\" id=\"Initialisieren\" value=\"Initialisieren\" onclick = \"javascript:return</pre></th></input<>	<pre>name=\"Initialisieren\" type=\"reset\" id=\"Initialisieren\" value=\"Initialisieren\" onclick = \"javascript:return</pre>				
		<pre>print(""):</pre>					
		<pre>print("");</pre>					
		<pre>print("");</pre>					
		if (isset(\$_POST['suc	:h'])) {				
		<pre>\$connectionstring = c</pre>	lb2_connect(\$system,				
			\$user,				
			<pre>\$passwd);</pre>				
		// Verbindungsfehler	abfangen und Script beenden				
		if (!\$connectionstrir	ng) {				
		die("Status: ". db2_	_conn_error(). " " .				
		db2_	<pre>_conn_errormsg());</pre>				
		}					
		if (\$_POST['GPTyp']==	= "KD") {				
		\$querv = "select					
		klidnr.					
		klname,					



Dialoge erstellen



Seite 39

klort, klstr, kltel, kldatn from epsw0402.svklst00 kl where upper(klname) like upper('%".\$\_POST['such']."%') order by klidnr"; \$query = "select klidnr, klname, where slkey = 'LAND' and klplz, klort, klstr, kltel, kldatn from epsw0402.svklst00 kl

```
(select slbez from epsw0402.svslst00
               where slkey = 'LAND' and
                     slwert = kl.kllandkz) as klland,
              klplz,
              klidnr between 1000000 and 6999999 AND
}
else {
              (select slbez from epsw0402.svslst00
                     slwert = kl.kllandkz) as klland,
```



```
Web-Programmierung
         Dialoge erstellen
5.2.5
                                   where
Seite 40
                                      klidnr >= 7000000 AND
                                      upper(klname) like
                                      upper('%".$_POST['such']."%')
                                   order by
                                      klidnr";
                        }
                         $queryexe = db2_exec($connectionstring, $query);
                        if (!$queryexe) {
                         die("Status: ". db2_stmt_error(). " " .
                                         db2_stmt_errormsg());
                        }
                         print("<TABLE width=\"95%\" border=\"0\">");
                           print("<TBODY>");
                             print("<TR>");
                               print("<TH>Identnr.</TH>");
                               print("<TH>Name</TH>");
                                      print("<TH>Land</TH>");
                                      print("<TH>PLZ</TH>");
                                      print("<TH>Ort</TH>");
                                      print("<TH>Strasse</TH>");
                                      print("<TH>Telefon</TH>");
                                      print("<TH>Kunde seit</TH>");
                                      print("</TR>");
                         while(db2_fetch_row($queryexe))
                         {
                                      $klidnr = db2_result($queryexe, 'KLIDNR');
```



Kapitel

	<pre>\$klname \$klland \$klplz \$klort \$klstr \$kltel \$kldatn</pre>	<pre>= db2_result(\$queryexe, = db2_result(\$queryexe, = db2_result(\$queryexe, = db2_result(\$queryexe, = db2_result(\$queryexe, = db2_result(\$queryexe, = db2_result(\$queryexe,</pre>	<pre>'KLNAME'); 'KLLAND'); 'KLPLZ'); 'KLORT'); 'KLSTR'); 'KLTEL'); 'KLDATN');</pre>	5.2.5 Seite 41									
<pre>print("<tr>"     print(     print(     print(     print(     print(     print(     print(     print(     print(     print("<th><pre>'); ("<td>\$k1id ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td></td></td></pre></th></tr>"); BODY&gt;"); BLE&gt;");</pre>	<pre>'); ("<td>\$k1id ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td></td></td></pre>	\$k1id (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("</td></td>	\$k1na (" <td>\$k1na ("</td>	\$k1na ("	<pre>Inr"); ume"); und"); z "); rt "); rt "); e1 "); utn");</pre>		
<pre>'); ("<td>\$k1id ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td></td></td></pre>	\$k1id (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("<td>\$k1na ("</td></td></td>	\$k1na (" <td>\$k1na ("<td>\$k1na ("</td></td>	\$k1na (" <td>\$k1na ("</td>	\$k1na ("				
<pre>print(" </pre>	ly>");												



Seite 42

## 5.2.5.13 Optimierungen 5: Ordnung schaffen

Wir sind bis jetzt schon sehr weit gekommen. Wenn Sie sich jedoch das komplette Script "svklst01.php" in der Version 1.1 ansehen, dann haben Sie wahrscheinlich Schwierigkeiten, sich dort zu orientieren.

Ein Bestandteil des Scripts besteht aus dem Element der Dialogsteuerung. Ein weiterer umfasst die Anzeigelogik. Ein dritter Bestandteil enthält die Abfragelogik für den Datenbankzugriff.

An dieser Stelle stellt sich die berechtigte Frage, ob wir diese drei Komponenten nicht voneinander trennen können. Die Antwort ist selbstverständlich: Ja.

Gehen wir von folgendem Konzept aus:

- 1. Es wird ein Master-Dokument "svklmaster.php" realisiert. Dieses Dokument enthält die Dialogsteuerung.
- 2. Es wird ein untergeordnetes Dokument "svkldisplay.php" realisiert, dass die Anzeigelogik des Suchformulars enthält.
- 3. Es wird ein weiteres untergeordnetes Dokument "svkldbaccess.php" realisiert, das die Datenbankzugriffslogik enthält.

## 5.2.5.13.1 Das Script SVKLMASTER.PHP - V1.0

Die Grundidee beim Script "svklmaster.php" besteht darin, dass die abhängigen Scripte "svkldisplay.php" und "svkldbaccess.php" via der Funktion "required() "in das Master-Dokument inkludiert werden.

Der Vorteil dieser Methode besteht darin, dass für andere, gleichartige Dialoge, dieses Master-Dokument nur kopiert werden muss, so das mir die generelle Dialoglogik für ein anderes Aufgabengebiet zur Verfügung steht.

Im Folgenden sehen Sie das Script "svklmaster.php":

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1" />
<LINK href="Styles/format.css" rel="stylesheet"
type="text/css">
<script src="js/formhandling.js"
```



apitel Web-Programmierung Dialoge erstellen type="text/javascript"></script> Seite 43 <title>Kunden und Lieferantenliste</title> </head> <?php require("includes/localSettings.php"); print("<body onload=\"javascript:return</pre> pageHandler();\">"); require("svkldisplay.php"); if (isset(\$\_POST['such'])) { require("svkldbaccess.php"); } print("</body>"); ?> </body> </html>

Ergänzung 4∕2008

Ein ITP Handbuch

Die relevanten Zeilen wurden von mir fett markiert.

**Dialoge erstellen** 

5.2.5 Seite 44

## 5.2.5.13.2 Das Script SVKLDISPLAY.PHP - V1.0

Das von "svklmaster.php" abhängige Script "svkldisplay" enthält nun nur noch die Anzeigelogik für das Suchformular.

```
<?php
```

```
print("<form id=\"sucheKL\" name=\"sucheKL\"</pre>
           method=\"post\" action=\"svklmaster.php\"
           onsubmit=\"javascript:return
                              validate(this)\"/>");
 print("");
  print("");
    print("Kunden und Lieferanten suchen");
    print(" ");
  print("");
  print("");
    print("Suchbegriff");
    print("<input name=\"such\"</pre>
                    type=\"text\"
                    value=\"".$_POST['such']."\"
                    id = \"such \"
                    size=\"60\"
                    maxlength=\"60\"/>");
  print("");
  print("");
    print("Kunden / Lieferanten");
    print("");
        print("<input name=\"GPTyp\"</pre>
                    type=\"radio\"
                    value=\"KD\"
                    checked=\"checked\"/>");
     if ($_POST['GPTyp'] == "LF"){
```



**Dialoge erstellen** 



Seite 45

print("<input name=\"GPTyp\"</pre> type=\"radio\" value="LF"checked=\"checked\"/>"); } else { print("<input name=\"GPTyp\"</pre> type=\"radio\" value="LF "/>");} print(""); print(""); print(""); print(" "); print(""); print("<input name=\"Suchen\"</pre> type=\"submit\" id="Suchen"value=\"Suchen\" />"); print("<input name=\"Initialisieren\"</pre> type=\"reset\" id=\"Initialisieren\" value=\"Initialisieren\" onclick = \"javascript:return initialize(sucheKL)\"/>"); print(""); print(""); print("</form>");

?>





Dialoge erstellen

5.2.5 Seite 46 Ein erster Test zeigt folgenden Dialog:

Kunden und Lieferanten s	uchen		
Suchbegriff			
Kunden / Lieferanten	c c		
	Suchen	Initialisieren	

"Svkldisplay.php" erfolgreich eingebunden

Anschließend binden wir das nächste Dokument ein.

## 5.2.5.13.3 Das Script SVKLDBACCESS.PHP – V1.0

Jetzt schauen wir uns das Script "svkldbaccess.php" an.



Dialoge erstellen

apitel

```
where slkey = 'LAND' and
                                                                       Seite 47
                     slwert = kl.kllandkz) as klland,
              klplz,
              klort,
              klstr,
              kltel,
              kldatn
           from
              epsw0402.svklst00 kl
           where
              klidnr between 1000000 and 6999999 AND
              upper(klname) like
              upper('%".$_POST['such']."%')
           order by
              klidnr";
}
else {
              $query = "select
              klidnr,
              klname,
              (select slbez from epsw0402.svslst00
               where slkey = 'LAND' and
                     slwert = kl.kllandkz) as klland,
              klplz,
              klort,
              klstr,
              kltel,
              kldatn
           from
              epsw0402.svklst00 kl
           where
```



```
Web-Programmierung
         Dialoge erstellen
5.2.5
                                      klidnr >= 7000000 AND
Seite 48
                                      upper(klname) like
                                      upper('%".$_POST['such']."%')
                                   order by
                                      klidnr";
                       }
                         $queryexe = db2_exec($connectionstring, $query);
                       if (!$queryexe) {
                        die("Status: ". db2_stmt_error(). " " .
                                         db2_stmt_errormsg());
                       }
                         print("<TABLE width=\"95%\" border=\"0\">");
                           print("<TBODY>");
                             print("<TR>");
                               print("<TH>Identnr.</TH>");
                               print("<TH>Name</TH>");
                                     print("<TH>Land</TH>");
                                     print("<TH>PLZ</TH>");
                                     print("<TH>Ort</TH>");
                                     print("<TH>Strasse</TH>");
                                     print("<TH>Telefon</TH>");
                                     print("<TH>Kunde seit</TH>");
                                     print("</TR>");
                        while(db2_fetch_row($queryexe))
                         {
                               $klidnr = db2_result($queryexe, 'KLIDNR');
                               $klname = db2_result($queryexe, 'KLNAME');
```



Kapitel



Anschließend testen wir die Funktionsfähigkeit:



Dialoge erstellen

## 5.2.5 Seite 50

Kunden u	nd Liereranten su	anen						
Suchbegrif	f	ds						
Kunden / L	.ieferanten	• •						
		Suchen Initialisieren						
dentnr.	Name		Land	PLZ	Ort	Strasse	Telefon	Kunde sei
014500	Kraft Foods Deutsd	hland GmbH & Co. KG	Deutschland	28199	Bremen	Langemarckstr. 4-20	0421/59901 e	2002-12-03
027400	Harrods Knightsbri	dge Limited	Groß Britannien	SWIX 7X	London	Knightsbridge	0044/1717301234	2002-05-29
042000	Berendsohn AG		Deutschland	22559	Hamburg	Alte Sülldorfer Landstr. 400	040/81909106 e	2002-05-29
064000	JD Edwards Deutso	hland GmbH	Deutschland	64546	Mörfelden/Frankfurt	Waldeckerstrasse 11	06103/7260 e	2002-12-11
089900	Wort+Bild Verlag K	Conradshöhe GmbH & Co.	Deutschland	82065	Baierbrunn	Konradshöhe	089/744330 e	2002-12-12
142600	ADS Informatik Ha	rd- u. Software-Beratung	Deutschland	66693	Mettlach	Goetheweg 8	06865/93300 e	2002-12-12
202600	Smith + Nephew R	ichards GmbH Medizintechnik	Deutschland	78532	Tuttlingen	Alemannenstr. 14	07462/2080 e	2003-02-11
205800	DIFA Deutsche Imr	mobilien Fonds AG	Deutschland	20354	Hamburg	Valentinskamp 20	040/349190 e	2002-05-29
224500	HUDSON Intern. V	ertr. GmbH	Deutschland	70565	Stuttgart	Am Wallgraben 45	0711/78150	2002-05-29
231200	Holert Kalksandstei	nwerke GmbH & Co.	Deutschland	24568	Kaltenkirchen	Barmstedter Str. 14	04191/7050 e	2002-05-2
257600	Fonds f. Umweltstu	idien e.V.	Deutschland	53175	Bonn	Godesberger Allee 108 - 112	0228/2692/0	2002-05-29
272200	IDS Inter Data		Deutschland	63477	Maintal	Am Kreuzstein 82-84	06109/76680	2002-05-29
311600	TDS Tele Columbu	s Daten und Service GmbH	Deutschland	30519	Hannover	Peiner Str. 8	0511/98765 e	2002-05-29
317100	Virgin Records Gm	bH	Deutschland	80803	München	Herzogstr. 64	089/381950	2002-05-2
317200	Gutehoffnungshütt	e Radsatz GmbH	Deutschland	46145	Oberhausen	Steinbrinkstr. 170	0208/7400/0 e	2002-05-2
335000	DSM Computer		Deutschland	57299	Burbach	Forstweg 14	02736/3176	2003-04-22
352900	IDS GmbH		Deutschland	69517	Gorxheimertal	Am Wetzelsberg 2	06201/292091 e	2003-04-15
353200	Berendsen PMC Gr	nbH Steinebronn Industriebedarf Geschäftsbereich	Deutschland	70499	Stuttgart	Kranstr. 9	0711/8361/0	2002-05-29
377900	Verbandsgemeinde	verwaltung EDV-Organisation	Deutschland	57610	Altenkirchen	Rathausstraße 13	02681/850	2002-05-29
380100	Harley-Davidson G	mbH	Deutschland	64546	Mörfelden-Walldorf	Starkenburgstr. 12	06105/2840	2002-05-29
385000	Feige, Siegfried LK	W-Instandsetzung	Deutschland	02708	Löbau	Weißenberger Str. 45	03585/862537 e	2002-10-09
434300	Mundschenk Druck	- und Verlags- gesellschaft mbH	Deutschland	29614	Soltau	Harburger Str. 63	05191/808/0 e	2002-02-18
464700	Unilever Best Food	s Deutschland GmbH	Deutschland	20355	Hamburg	Dammtorwall 15	040/3493/0 e	2002-02-06
483200	HDS Rechenzentru	m GmbH & Co. KG	Deutschland	49577	Ankum	Am Lordsee 1	05462/765/0 e	2002-08-05
498400	Randstad Organisa	tion für Zeitarbeit GmbH	Deutschland	65760	Eschborn	Ginheimerstr. 5	06196/408/0 e	2003-01-09
032801	Deutsche Immobili	en Fonds AG DIFA	Deutschland	20355	Hamburg	Caffamacherreihe 8	040/34919-0 e	2003-01-14
045201	WestInvest Gesells	chaft für Investmentfonds mbH	Deutschland	40476	Düsseldorf	Hans-Böckler-Str. 33	0211/51798-0 e	2002-06-05
054401	Fonds für Umwelts	tudien e.V.	Deutschland	53175	Bonn	Godesbergerallee 108-112	0228/2692216 e	2003-06-07
054701	EDC Informationet	achnologie & Service GmbH	Doutschland	53220	Bonn	Kautovetr 52	0228/488-0 e	2002-11-00

#### Der Dialog steht!

Nach erfolgreichem Funktionstest könnten wir uns eigentlich zufrieden zurücklehnen. Allerdings sollten wir uns noch einmal vergegenwärtigen, was wir bisher erreicht haben.

Wir können jetzt verschiedene Funktionsbereiche unseres Dialogs gezielt ansprechen.

#### **Positiv:**

Wenn wir Probleme in der Dialogfolge haben (zum Beispiel: wenn das Formular zwar angezeigt wird, aber es erfolgt keine Anzeige der Datentabelle nach einer Sucheingabe), so brauchen wir uns lediglich in das Script "svklmaster. php" hineinzubegeben, um einen Fehler zu analysieren und zu beseitigen.

Wenn wir Probleme mit dem Suchformular haben, so wissen wir, das Problem liegt im Script "svkldisplay.php".

Wenn die Daten in der Tabelle nicht angezeigt werden, dann können wir davon ausgehen, dass der Fehler in der Datenbankzugriffslogik liegt. Wir müssen also das Script "svkldbaccess.php" korrigieren.

#### Negativ:

Mit Ausnahme des Scripts "svklmaster.php" haben wir immer noch eine hohe Funktionsdichte innerhalb eines Scripts, so dass eine Problembereinigung in "svkldisplay.php" und "svkldbaccess.php" nicht unbedingt banal ist.

- · Wie können wir hier eine bessere Wartung und Fehlererkennung erreichen?
- Wie können wir erreichen, dass einmal geschriebene Funktionen in weitere sachbezogene Problemstellungen übernommen werden können?

Um dies zu beantworten, müssen wir das Script "svkldbacces.php" genauer durchleuchten.



# **L** Kapitel

## 5.2.5.14 PHP-Funktionen einsetzen

Wenn wir das Script svkldbaccess.php einmal genauer betrachten, so lassen sich vier Funktionsbereiche isoliert betrachten:

- Verbindung zur Datenbank herstellen
- Abfrage ausführen
- Daten anzeigen
- Verbindung zur Datenbank beenden

Diese vier Funktionsbereiche werden wir jetzt getrennt betrachten und in so genannten PHP-Funktionen darstellen.

## 5.2.5.14.1 PHP Funktionen

Neben der Möglichkeit in PHP eingebaute Funktionen zu nutzen, existiert die Möglichkeit eigene Funktionen zu erstellen. Da Sie sich bereits mit JS-Funktionen beschäftigt haben, dürfte Ihnen dieses Kapitel nicht schwer fallen, dass die Analogien zwischen beiden Syntaxen die Unterschiede überwiegen.

```
<?php
function name()
{
// ... Die Befehle
}
?>
```

Eine Funktionsdefinition wird mit dem Schlüsselwort function eingeleitet. Danach kommt der Name der Funktion. Dieser darf nicht mit einer Zahl anfangen. Nach dem ersten Zeichen dürfen alphanumerische Zeichen und Unterstriche benutzt werden. Danach kommen runde Klammern, in denen dann Variablen für Parameter stehen. Doch dazu kommen wir gleich. Oben im Beispiel haben wir angegeben, dass diese Funktion keine Parameter bekommt. Dann kommt ein Programmblock in geschweiften Klammern, in denen dann die Befehle stehen, die dann ausgeführt werden. Dies könnte dann so aussehen.



Ergänzung 9/2009

5.2.5 Seite 51



Dialoge erstellen

5.2.5 Seite 52

```
<?php
function hello()
{
print("Hallo, ich wurde von einer Funktion
ausgegeben");
}
?>
```

Diese Funktion kann man dann mit hello() aufrufen.

```
<?php
function hello()
{
    print("Hallo, ich wurde von einer Funktion
ausgegeben");
}
hello(); // Aufruf der Funktion
?>
```

Wenn wir die Funktion aufrufen, werden die enthaltenen Befehle ausgeführt und in diesem Fall wird die print() – Funktion ausgeführt.





#### 5.2.5.14.2 Parameter

Mit Parametern können wir die Funktionsweise einer Funktion steuern bzw. die Funktion kann auf die Parameter reagieren und entsprechende Aktionen ausführen.

Um Parameter zu definieren, schreibt man Variablen in den runden Klammern, getrennt durch Kommata.

Dies sieht dann so aus:

```
<?php
function xyz($param1, $param2)
{
// Anweisungen hier
}
?>
```

Wenn jetzt diese Funktion mit zwei Parametern aufgerufen wird, enthält die Variable \$param1 den Wert des ersten Parameters und \$param2 den Wert des zweiten Parameters. Diese Variablen können dann im Funktionsteil verwendet werden.

In einem Beispiel würde das jetzt so aussehen.

```
<?php
function xyz($param1, $param2)
{
    print("Der 1. Parameter ist
".$param1."
und der 2. ist
".$param2");
}
xyz("Wert", 200);
?>
```

Dies erzeugt folgende Ausgabe.

Der 1. Parameter ist Wert und der 2. ist 200

Die gewählten Variablennamen \$param1 und \$param2 müssen nicht so heißen. Sie sollten aber eindeutige Namen haben.



5.2.5 Seite 53

## 5.2.5 Seite 54

#### 5.2.5.14.3 Gültigkeit von Variablen

Wenn eine Funktion aufgerufen wird, kennt sie nur ganz bestimmte Variablen. Einmal kennt sie die Variablen, die als Parameter übergeben wurden. Dann kennt sie die sogenannten Superglobals. Dies sind die Variablen \$\_GET, \$\_ POST, \$\_REQUEST, \$\_SERVER, \$\_ENV, \$\_COOKIE und \$\_SESSION, von denen Sie bisher \$\_POST kennengelernt haben. Und dann kennt sie noch die Variable \$GLOBALS.

Alle anderen Variablen, z.B. aus dem Hauptscript, sind der Funktion unbekannt.

```
<?php
function xyz()
{
print($var1);
}
$var1 = "Fehler";
xyz();
?>
```

Wenn man dieses Script startet, wird PHP eine Nachricht ausgeben, dass die Variable \$var1 nicht definiert ist.

Diese Lokalität der Variablen gilt auch umgekehrt. Eine Variable in einer Funktion ist nicht im Hauptprogramm zu erreichen (es sei denn, sie wird zurückgegeben, doch dazu kommen wir gleich).

Wie könnte diese Funktion aussehen, damit die Zeichenfolge in varl ausgegeben wird?

```
<?php
function xyz($param){
$localvar = $param
print($localvar);
}
$var1 = "Korrekt";
xyz($var1);
?>
```

In diesem Fall wird \$var1 an die Funktion xyz als Parameter übergeben. \$var1 wird im parameter \$param empfangen.



Web-Programmierung	bit d
Dialoge erstellen	Ka
Der Inhalt von \$param wird an die nur in der Funktion xyz sichtbare Variable \$localvar übergeben und dann von der print()-Funktion ausgegeben. \$localvar ist im Hauptscript nicht sichtbar und kann hier auch nicht verwendet werden.	5.2.5 Seite 55

## 5.2.5.14.4 Rückgabewerte

Wenn wir einen Wert aus einer Funktion zurückgeben wollen, müssen wir den Ausdruck return benutzen.

Mit return \$Wert; können wir aus einer Funktion heraus einen Wert zurückgeben. Diesen Wert können wir dann in eine Variable speichern oder direkt verarbeiten, z. B. mit print() ausgeben oder mit if verarbeiten.

Wenn an einer Stelle ein return steht, so wird die Funktion an dieser Stelle beendet und der Wert (falls angegeben) zurückgegeben.

Befehle, die hinter dem return-Befehl stehen, werden nicht verarbeitet.

0



## 5.2.5 Seite 56

#### 5.2.5.14.5 Optionale Parameter

Auch optionale Parameterübergaben sind in PHP realisierbar.

Hinter dem zweiten Parameter sehen Sie eine Wertzuweisung. Dieses heißt, wenn der zweite Parameter nicht übergeben wird, wird die Variable des zweiten Parameters mit diesem Wert gefüllt. Auf diesen Wert können wir dann auch reagieren.

Folgende Aufrufe der Funktion xyz sind möglich:



# **Z** Kapitel

## 5.2.5.15 Datenbankzugriffsfunktionen erstellen

In Kapitel 5.2.5.14 hatten wir bemerkt, dass wir vier Funktionsbereiche beim Datenbankzugriff voneinander isolieren können.

Ich rekapituliere:

- Verbindung zur Datenbank herstellen
- Abfrage ausführen
- Daten anzeigen
- Verbindung zur Datenbank beenden

Für alle Funktion gilt, dass sie wieder in einem einzelnen Script db2functions. php zur Verfügung gestellt werden.

## 5.2.5.15.1 Die Funktion ConnectToDB2()

Im Folgenden beabsichtige ich die Funktion ConnectToDB2 zu erstellen, die den db2\_Connect() kapselt. Die Merkmale dieser Funktion sind:

- Verbindung zu jeglicher DB2 herstellen
- Fehlermeldung erzeugen, falls connect nicht erfolgreich ist
- Datenbank-Handle an aufrufenden Prozess zurückgeben.

Ich erzeuge ein neues Dokument mit dem Namen db2functions.php im Verzeichnis /includes.

Die Funktion ConnectToDB2() wurde wie folgt kodiert:

php</th <th></th> <th></th> <th></th> <th></th> <th></th>					
function	<pre>ConnectToDB2(\$DB2,</pre>	\$DBuser,	<pre>\$DBpasswd)</pre>	{	

Die Funktion erwartet drei Parameter

- Name der Datenbank
- Benutzer der Datenbank
- Kennwort des Datenbankbenutzers

Ergänzung 9/2009





Dialoge erstellen

5.2.5 Seite 58 Die Verbindung zur Datenbank wird hergestellt, dazu werden die Parameterwerte an die Funktion db2\_connect() übergeben.

Kann die Verbindung zur Datenbank nicht erstellt werden, so wird die gesamte Verarbeitung abgebrochen und eine Fehlermeldung ausgegeben.

return \$DBHandle;
}

Wird die Verbindung zur Datenbank hergestellt, dann wird der Verbindungshandle an den aufrufenden Prozess zurückgegeben

?>

Hier noch einmal die unkommentierte Funktion:

```
function ConnectToDB2($DB2, $DBuser, $DBpasswd) {
  $DBHandle = db2_connect($DB2,
  $DBuser,
  $DBpasswd);
  if (!$DBHandle) {
   die("Status: ". db2_conn_error(). " " .
   db2_conn_errormsg());
  }
  return $DBHandle;
  }
```

Der Aufruf der Funktion ConnectToDB2() aus dem Script svkldbaccess.php sieht dann folgendermaßen aus:

```
$connectionstring = ConnectToDB2($system,
$user,
$passwd);
```

Testen Sie die Funktion.





## 5.2.5.15.2 Die Funktion ExecuteDB2Sql()

Jetzt realisieren wir die Ausführung des SQL-Befehls auf dem System i. Die Funktion ExecuteDB2Sql() soll folgende Merkmale aufweisen.

Mittels eines Eingabeparameters wird entschieden, ob Kunden oder Lieferantensätze abgefragt werden. Übergeben wir den Wert "KD", so werden Kunden abgefragt, übergeben wir "LF", so erhalten wir Lieferantensätze.

Ein weiterer Eingabeparameter ist der Suchbegriff.

Kommt es bei der Abfrage zu einem Fehler, dann soll eine Fehlermeldung ausgegeben werden und die Verarbeitung wird komplett beendet.

Bei erfolgreicher Ausführung wird der entsprechende Resultset an den aufrufenden Prozess zurückgegeben

Ich kodiere die Funktion ExecuteDB2Sql():

An die Funktion werden die drei Parameter Datenbankhandle, Selectionskriterium (Kunden/Lieferanten) und der Suchbegriff übergeben.

```
if ($Selection== "KD") {
        $query = "select
                      klidnr,
                     klname,
                      (select slbez from epsw0402.svslst00
                      where slkey = 'LAND' and
                             slwert = kl.kllandkz) as klland,
                      klplz,
                      klort.
                      klstr,
                      kltel.
                     kldatn
                  from
                     epsw0402.svklst00 kl
                  where
                      klidnr between 1000000 and 6999999 AND
                     upper(klname) like
```





```
upper('%".$Searchargument."%')
   }
   else {
        $query = "select
                  klidnr,
                  klname,
                   (select slbez from epsw0402.svslst00
                   where slkey = 'LAND' and
                          slwert = kl.kllandkz) as klland,
                   klplz,
                   klort,
                   klstr,
                   kltel,
                   kldatn
                from
                   epsw0402.svklst00 kl
                where
                   klidnr >= 7000000 AND
                   upper(klname) like
                   upper('%".$Searchargument."%')
                order by
                   klidnr";
}
   $ResultSet = db2_exec($DBHandle, $query);
   if (!$ResultSet) {
     die("Status: ". db2_stmt_error(). " " .
                      db2_stmt_errormsg());
   }
    return $ResultSet;
```

An den aufrufenden Prozess wird der Resultset der Abfrage zurückgegeben.

}

Ich verzichte auf die Erläuterung der sonstigen Funktionen, da sie fast 1:1 aus dem Script svkldbaccess.php übernommen wurden.





## 5.2.5.15.3 Die Funktion DisplayDB2ResultSet()

Der nächste Schritt besteht darin die abgefragten Daten anzuzeigen. Dazu werde ich die Funktion DisplayDB2ResultSet() realisieren.

Die Funktion hat folgende Merkmale:

- Es wird der Resultset, den die Funktion ExecuteDB2Sql() erzeugt hat, empfangen
- Die Funktion soll die Anzahl der gefundenen Sätze zurückgeben.
- · Sollte der Resultset gefüllt sein, dann soll er ausgegeben werden

Hier der Funktionscode:

```
function DisplayDB2ResultSet($ResultSet) {
   RowCount = 0;
    print("<TABLE width=\"95%\" border=\"0\">");
      print("<TBODY>");
        print("<TR>");
          print("<TH>Identnr.</TH>");
          print("<TH>Name</TH>");
            print("<TH>Land</TH>");
            print("<TH>PLZ</TH>");
            print("<TH>Ort</TH>");
            print("<TH>Strasse</TH>");
            print("<TH>Telefon</TH>");
            print("<TH>Kunde seit</TH>");
          print("</TR>");
   while(db2_fetch_row($ResultSet))
    {
      $RowCount = $RowCount + 1;
      $klidnr = db2_result($ResultSet, 'KLIDNR');
      $klname = db2_result($ResultSet, 'KLNAME');
      $klland = db2_result($ResultSet, 'KLLAND');
      $klplz
                = db2_result($ResultSet, 'KLPLZ');
```



Ergänzung 9/2009

5.2.5 Seite 61

<b>L</b> bite	Web-Prog	<b>Jrammierung</b>		
Ka	Dialoge ers	tellen		
5.2.5 Seite 62		<pre>\$klort \$klstr \$kltel \$kldatn print("&lt; print } print("</pre>	<pre>= db2_result(\$ResultSet, = db2_result(\$ResultSet, = db2_result(\$ResultSet, = db2_result(\$ResultSet, = db2_result(\$ResultSet,</pre>	<pre>'KLORT'); 'KLSTR'); 'KLTEL'); 'KLDATN');</pre>
		}		

Eine Erläuterung dieser Funktion erübrigt sich, da Sie die Logik aus den bisher entwickelten Scripten kennen. Beachten Sie jedoch die lokale Variable \$RowCount, die an das aufrufende Script zurückgegeben wird.

## 5.2.5.15.4 Die Funktion CloseDB2Connection()

Die letzte Funktion CloseDB2Connection() wird dann die Datenbankverbindung beenden.

Diese Funktion wird den Datenbankhandle empfangen und die Datenbankverbindung schließen.

Ich halte die Funktionalität dieser Funktion für leicht verständlich und werde daher den Code nicht kommentieren.

```
function CloseDB2Connection($DBHandle) {
  db2_close($DBHandle);
}
```





Seite 63

## 5.2.5.15.5 Das Script SVKLDBACCESS.PHP V1.1

Wenden wir uns jetzt dem Aufruf der Funktionen zu. Hier sehen Sie den Code für svkldbacces.php in der neuen Version V1.1:

```
<?php
require("includes/db2functions.php");
$connectionstring = ConnectToDB2($system,
$user,
$passwd);
$queryexe = ExecuteDB2Sql($connectionstring,
$_POST['GPTyp'],
$_POST['Such']);
if (DisplayDB2ResultSet($queryexe)== 0) {
print("Es wurden keine Sätze gefunden.");
}
CloseDB2Connection($connectionstring);
?>
```

Achten Sie auf die Funktion require(), die das Script db2functions.php anfordert.



Kapitel

Web-Programmierung

Dialoge erstellen

# 5.2.55.2.5.15.6Seite 64Das Script db2functions.php

```
<?php
   function ConnectToDB2($DB2, $DBuser, $DBpasswd) {
      $DBHandle = db2_connect($DB2,
                                       $DBuser,
                                       $DBpasswd);
      if (!$DBHandle) {
         die("Status: ". db2_conn_error(). " " .
                         db2_conn_errormsg());
      }
      return $DBHandle;
  }
  function ExecuteDB2Sql($DBHandle,
                          $Selection,
                          $Searchargument) {
      if ($Selection== "KD") {
        $query = "select
                     klidnr,
                     klname,
                     (select slbez from epsw0402.svslst00
                      where slkey = 'LAND' and
                            slwert = kl.kllandkz) as klland,
                     klplz,
                     klort,
                     klstr,
                     kltel,
                     kldatn
                  from
                     epsw0402.svklst00 kl
                  where
                     klidnr between 1000000 and 6999999 AND
```



Dialoge erstellen



```
upper(klname) like
                  upper('%".$Searchargument."%')
               order by
                  klidnr";
   }
   else {
        $query = "select
                  klidnr,
                  klname,
                   (select slbez from epsw0402.svslst00
                   where slkey = 'LAND' and
                          slwert = kl.kllandkz) as klland,
                   klplz,
                   klort,
                   klstr,
                   kltel,
                   kldatn
                from
                   epsw0402.svklst00 kl
                where
                   klidnr >= 7000000 AND
                   upper(klname) like
                   upper('%".$Searchargument."%')
                order by
                   klidnr";
}
   $ResultSet = db2_exec($DBHandle, $query);
   if (!$ResultSet) {
     die("Status: ". db2_stmt_error(). " " .
                     db2_stmt_errormsg());
   }
   return $ResultSet;
```





}

Dialoge erstellen

## 5.2.5 Seite 66

```
function DisplayDB2ResultSet($ResultSet) {
RowCount = 0;
print("<TABLE width=\"95%\" border=\"0\">");
   print("<TBODY>");
    print("<TR>");
      print("<TH>Identnr.</TH>");
       print("<TH>Name</TH>");
         print("<TH>Land</TH>");
         print("<TH>PLZ</TH>");
         print("<TH>Ort</TH>");
         print("<TH>Strasse</TH>");
         print("<TH>Telefon</TH>");
         print("<TH>Kunde seit</TH>");
       print("</TR>");
while(db2_fetch_row($ResultSet))
 {
   $RowCount = $RowCount + 1;
   $klidnr = db2_result($ResultSet, 'KLIDNR');
   $klname = db2_result($ResultSet, 'KLNAME');
   $klland = db2_result($ResultSet, 'KLLAND');
   $klplz = db2_result($ResultSet, 'KLPLZ');
            = db2_result($ResultSet, 'KLORT');
   $klort
   $klstr = db2_result($ResultSet, 'KLSTR');
            = db2_result($ResultSet, 'KLTEL');
   $kltel
   $kldatn = db2_result($ResultSet, 'KLDATN');
   print("<TR>");
       print("<TD>$klidnr</TD>");
```



	Web-Programmierung Dialoge erstellen	<b>P</b>				
<pre>print("<td>\$klname</td>"); print("<td>\$klland</td>"); print("<td>\$klplz </td>"); print("<td>\$klort </td>"); print("<td>\$klstr </td>"); print("<td>\$kltel </td>"); print("</pre>	\$klname	\$klland	\$klplz	\$klort	\$klstr	\$kltel

");  
return \$RowCount;  
}  
function CloseDB2Connection(\$DBHandle) {  
db2\_close(\$DBHandle);  
}  | 5.2.5 Seite 67 ||  |  |  |


## 5.2.5 Seite 68

#### 5.2.5.16 Das Formular optimieren

Wenn Sie einmal Bearbeitungsformulare vor Ihrem geistigen Auge Revue passieren lassen, so ist sicherlich auffällig, dass ein Formular aus einer häufigen Wiederholung gleicher Eingabeelemente besteht. Aus diesem Grund bieten sich Formulare für Optimierungen geradezu an.

Denken wir doch einmal an Eingabefelder. Was unterscheidet sie voneinander? Zum Beispiel die Feldlänge, der interne Name – aber dann? Ziehen wir weiterhin ins Kalkül, dass wir nicht nur in einem Dialog Eingabefelder benötigen, sondern in vielen, dann würde das ständige Kodieren von Eingabefeldern in HTM bzw. PHP irgendwann zur Strafarbeit mutieren, auch dann, wenn man HTML Generatoren wie WDSC und/oder Dreamweaver zu Hilfe nehmen würden.

Grundsätzlich könnte ich hier mit den gleichen Mitteln verfahren wie bei der Optimierung der Datenbankzugriffe, indem ich eine Funktion erstelle, die z. B. Eingabefelder ausgibt. und über eine Parameterleiste steuere ich die Attributierung des Feldes.

Ich möchte Ihnen hier jedoch eine alternative Methode vorstellen, die auf einem Objekt-orientierten (OO)Ansatz basiert und Ihnen einen Ansatz liefern soll in PHP einen OO-basierten Programmierungsansatz zu finden.

Bitte missverstehen Sie mich nicht. Ich werde hier mit Sicherheit keinen Kurs in OOP (objekt-orientierter Programmierung) abliefern. Dafür gibt es Literatur und auch genug im Internet. Aber einen Denkansatz möchte ich Ihnen schon liefern.



## 5.2.5.16.1 Exkurs: Was ist OOP?

OOP steht für Objekt-orientierte Programmierung. Was bedeutet das? Nun, bisher haben wir in unseren Variablen nur Werte gespeichert, Funktionen aufgerufen und Rückgabewerte von Funktionen in Variablen gespeichert.

Im Grunde besteht ein Programm aus drei Teilen: Aus Anweisungen (mittels Funktionen und Operatoren), aus Variablen und aus Werten. Die Programmierweise, die wir kennengelernt haben, nennt sich imperative Programmierung, bzw. – dank des Fehlens einer "GOTO"-Anweisung – strukturierte Programmierung.

Wieso sollten wir etwas daran ändern? Nun, zum einen ist diese Programmierweise mehr an Computern als an der wirklichen Welt orientiert. Ein Problem aus dem wirklichen Leben muss meist erst übersetzt werden, damit man es lösen kann. Zum anderen ist Code, der Objekt-orientiert geschrieben ist, eher wiederverwendbar als nicht-oo Code. Wieso?

[Beispiel aus "Programming Ruby" von David Thomas, Andrew Hunt] Nun, angenommen wir haben eine Funktionensammlung für Mathematik/Trigonometrie, die wir brauchen um geometrische Probleme zu lösen. Diese enthält -natürlich- auch eine Funktion namens "sin", um den Sinus eines Winkels zu berechnen. Eine andere Funktionensammlung beschäftigt sich mit Gut und Böse und hat auch eine Funktion namens "sin", die benutzt wird um die Anzahl der Sünden eines Wesens herauszufinden. Ein Wesen mit 0 Sünden muss ein Engel sein. Getrennt funktionieren die beiden natürlich super, aber sobald dir jemand die Aufgabe stellt, zu berechnen, wie viele Engel auf eine Nadelspitze passen, hast du ein echtes Problem. Was tun wir dagegen? Wir programmieren Objekt-orientiert.

Anstatt einer Funktion "sin", die uns die Anzahl Sünden eines Wesens zurückliefert, behandeln wir die Wesen als Objekte und fragen sie, wie viele Sünden sie begangen haben.

Ein objektorientiertes Programm besteht aus Objekten, die miteinander kommunizieren. Kleines Beispiel: Nehmen wir an, ich hätte eine Katze mit dem Namen Mieke, die über telepathische Kräfte verfügt, die mich direkt beeinflussen könnten, würde es – das Objekt "Mieke", sobald es hungrig ist, eine Nachricht mit dem Inhalt "hole Futter" an mich – das Objekt Luttkus senden. Dann verarbeite ich diese Nachricht, indem ich Futter hole, und gebe dann Mieke die Antwort auf ihre Nachricht zurück, nämlich das angeforderte Futter.

Soweit, so gut. Aber wie läuft die Kommunikation zwischen den Objekten ab? Die Antwort lautet: Mit Methoden.



**Ein ITP Handbuch** 

## 5.2.5 Seite 70

## 5.2.5.16.2 Methoden

Methoden bestimmen, was für Nachrichten ein Objekt empfangen kann und wie es mit den Nachrichten umgeht. In dem genannten Beispiel mit der Katze wäre "hole Futter" eine Methode des Objektes Luttkus, "öffne dich!" wäre eine Methode des Objektes "Kühlschranktür" und so weiter. Nachrichten müssen aber nicht so simpel sein: Wenn Mieke wählerisch ist und Fisch als Futter will, wird sie die Nachricht "hole Futter Fisch" an mich senden (Die Katze Mieke hält mich für geistig stark eingeschränkt und denkt, dass mich zusätzliche Worte verwirren würden.) Und schon haben wir unser erstes Problem: Was für eine Methode soll diese Nachricht nun verarbeiten? Es könnte ja sein, dass es eine Methode namens "hole" gibt. Soll dann diese Methode mit den Zusatzinfos "Futter" und "Fisch" aufgerufen werden, oder die Methode "hole Futter" mit der Zusatzinfo ""Fisch"?

Man könnte jetzt ein obskures Trennzeichen vereinbaren, in PHP jedoch gelten für Methoden die gleichen Regeln wie für Funktionen, sprich: Der Name besteht aus Buchstaben, Zahlen und dem Underscore, beginnend mit einem Buchstaben oder einem Underscore ("\_"), die Zusatzinfos werden mit einer (eventuell auch leeren) Liste von Parametern übergeben. Es gibt die Übereinkunft, dass Methoden mit einem Kleinbuchstaben (oder einem Underscore, darauf komme ich später zurück) beginnen und jedes neue Wort durch einen Grossbuchstaben gekennzeichnet ist, zum Beispiel würde die Methode, die Mieke aufruft, "hole-Futter" heißen.

Bestehen Objekte nur aus Methoden? Nun, wenn das so wäre, könnte Mieke sich nicht merken, dass sie gerade Futter bekommen hat und jetzt satt ist, und würde mich dauernd Futter holen schicken – nicht gerade angenehm für mich. Objekte brauchen also ein "Gedächtnis", einen Zustand. Dieser Zustand setzt sich aus den Eigenschaften des Objektes zusammen.


#### 5.2.5.16.3 Eigenschaften

Eigenschaften sind wie gesagt das Gedächtnis eines Objekts. Mieke hätte z.B. eine Eigenschaft "hunger", die festhält, wie hungrig sie ist. Die Tür hätte eine Eigenschaft, in der gespeichert ist, ob sie offen ist, und so weiter.

Wichtig ist: Eigenschaften sind **interne** Bestandteile, man liest sie nicht von außen aus, und man überschreibt sie niemals von außen, auch wenn es – wie in PHP – möglich ist. Allein die Methoden eines Objektes bilden die Schnittstelle zur Außenwelt, nur über sie kommunizieren Objekte.

Wie wir gesehen haben, sind Objekte eine Kombination aus Methoden und einem Zustand. Die Methoden und den Anfangszustand für jedes Objekt einzeln zu definieren, ist ein bisschen aufwendig, wenn wir berücksichtigen, wie viele Katzen, Türen und Menschen es auf der Erde gibt. Wir brauchen also ein Grundgerüst für Objekte, welches die Methoden und eventuell auch Standardwerte für die Eigenschaften definiert. Diese Grundgerüste nennen sich Klassen.

#### 5.2.5.16.4 Klassen und Vererbung

Jedes Objekt hat genau eine Klasse. Die Katze Mieke hätte die Klasse "Katze", die Tür zum Kühlschrank hätte die Klasse "Tür" und ich hätte die Klasse "Mensch". Anstelle dessen sagen wir auch: Mieke ist eine Instanz der Klasse Katze. Instanzen einer Klasse haben immer die gleichen Methoden, verstehen also die gleiche Art von Nachrichten, aber sie haben nicht zwingend den gleichen Zustand, das heißt zwei verschiedene Instanzen der gleichen Klasse können auf die gleiche Nachricht durchaus unterschiedlich reagieren.

Es gibt ein weiteres praktisches Prinzip, das sich Vererbung nennt. Wenn wir eine Klasse erweitern wollen, müssen wir sie nicht komplett neu schreiben. Wenn wir Katzen haben wollen, die schwanger werden können, wäre es praktisch, wenn wir zwischen männlichen und weiblichen Katzen unterscheiden könnten, denn sonst könnte es uns passieren, dass auf einmal die männlichen Katzen schwanger werden – was nicht wirklich Sinn der Sache ist. Also erstellen wir zwei Unterklassen von "Katze" namens "Katze\_maennlich" und "Katze\_weiblich" und definieren in ihnen nur die Methoden zum schwängern und schwanger werden, und die Eigenschaften die dafür nötig sind – es wäre ja dumm, wenn die weibliche Katze nicht wüsste, dass sie schwanger ist. Instanzen der Klassen "Katze\_maennlich" und "Katze\_weiblich" können dann trotzdem alle Nachrichten verarbeiten, die eine Instanz der Klasse "Katze" verarbeiten kann.

Das war jetzt nur ein kleines bisschen Theorie über OOP, und, wie ich hoffe, "einsteigerfreundlich" geschrieben. Kommen wir jetzt zurück zu unserem Suchformular...





5.2.5 Seite 72

#### 5.2.5.16.5 Klassen erstellen

Im ersten Beispiel geht es darum, eine Klasse InputElementField zu erstellen, in der Eingabefelder ausgegeben werden. Zu diesem Zweck erstelle ich im Pfad / classes meines Projekts die Datei InputElementField.php.

Bevor wir Objekte nutzen können, müssen wir eine Klasse definieren. Dies geschieht in PHP mit dem Schlüsselwort "class". Eigenschaften deklariert man mittels "var". Methoden werden genau wie Funktionen deklariert, nur innerhalb einer Klasse.

```
<?php
class InputElementField {
    // Eigenschaften
    var $idname;
    var $size;
    var $maxlength;
    var $value;
    var $tabindex;
    var $mode;</pre>
```

Um meine Eingabefelder zu attributieren habe ich der Klasse InputElement-Field folgende Eigenschaften zugeordnet:

\$idname	<b>→</b>	Name des Eingabefelds
\$size	<b>→</b>	Länge des Feldes
\$maxlength	<b>→</b>	Maximal angezeigte Länge
\$value	<b>→</b>	Wert im Feld
\$tabindex	<b>→</b>	Reihefolge im Dialog
\$mode	<b>→</b>	Eingabefähig oder nicht

Diese Eigenschaften werden auch als Klassenvariablen bezeichnet.







Hier sehen Sie den so genannten Konstruktor einer Klasse. Der Konstruktor ist eine besondere Methode. Sie trägt den gleichen Namen wie die Klasse und wird automatisch ausgeführt, wenn eine neue Instanz der Klasse erstellt wird.

Der Konstruktor unserer Klasse empfängt die Parameter und gibt sie als Parameter an Set-Methoden weiter. Das Schlüsselwort \$this-> beschreibt, dass sich diese Set-Methoden in der gleichen Klasse befinden, wie z. B. der Konstruktor.



```
Kapitel
```

```
Web-Programmierung
```

5.2.5 Seite 74

```
// Set-Methoden
function setIdName($NewName) {
   $this->idname = $NewName;
   return $this->idname;
}
function setSize($NewSize) {
   $this->size = $NewSize;
   return $this->size;
}
function setMaxlength($NewMaxlength) {
   $this->maxlength = $NewMaxlength;
   return $this->maxlength;
}
function setValue($NewValue) {
   $this->value = $NewValue;
   return $this->value;
}
function setTabindex($NewTabindex) {
   $this->tabindex = $Tabindex;
   return $this->tabindex;
}
function setMode($NewMode) {
   $this->mode = $NewMode;
   return $this->mode;
}
```

Die Aufgabe der Set-Methoden besteht jetzt darin, die vom Konstruktor empfangenen Parameterwerte in die Klassenvariablen zu stellen.





Seite 75

function displayInputText(){
 print(,,<input
name=\"".\$this->idname."\"
 type=\"text\"
 id=\"".\$this->idname."\"
 size=\"".\$this->size."\"
 maxlength=\"".\$this->maxlength."\"
 value=\"".\$this->value."\"
 tabindex=\"".\$this->tabindex."\"
 \$this->mode
 />");
}

Mit der Methode displayInputText() wird das Eingabefeld angezeigt.





## 5.2.5 Seite 76

#### 5.2.5.16.6 Aufruf der Klasse

Hier sehen Sie jetzt die Anwendung der Klasse im PHP-Script SVKLDISPLAY. PHP

```
require("classes/InputElementField.php");
```

Damit dem Script SVKLDISPLAY.PHP die Existenz der Klasse InputElement-Field bekannt ist, wird dieses Script mit require() eingebunden.

<pre>\$Suchbegriff = new InputElementField(</pre>	
	"such",
	60,
	60,
	<pre>\$_POST['such'],</pre>
	1,
	"");

Zuerst erstellen wir eine neue Instanz der Klasse InputElementField mit new. Die Instanz trägt den Namen \$Suchbegriff. Bei der Instantiierung der Klasse InputElementField wird automatisch der Konstruktor aufgerufen. In der Klammer sehen Sie die Parameterübergabe.

\$idname	←	"such"
\$size	←	60
\$maxlength	+	60
\$value	+	<pre>\$_POST['such']</pre>
\$tabindex	+	1
\$mode	←	"" (leer → eingabefähig)

\$Suchbegriff->displayInputText();

Dann wird die Methode displayInputText() der Klasseninstanz \$Suchbegriff aufgerufen und das Eingabefeld wird mit den genannten Eigenschaften angezeigt.





Seite 77

#### 5.2.5.16.7 Das Script: InputElementField.php

```
<?php
 class InputElementField {
      // Eigenschaften
      var $idname;
      var $size;
      var $maxlength;
      var $value;
      var $tabindex;
      var $mode;
      // Der Konstruktor
      function InputElementField($idname,
                                  $size,
                                  $maxlength,
                                  $value,
                                  $tabindex,
                                  $mode){
          $this->setIdName($idname);
          $this->setSize($size);
           $this->setMaxlength($maxlength);
          $this->setValue($value);
          $this->setTabindex($tabindex);
          $this->setMode($mode);
      }
      // Set-Methoden
      function setIdName($NewName) {
```



Ergänzung 9/2009



# Web-Programmierung

**Dialoge erstellen** 

# 5.2.5 Seite 78

```
$this->idname = $NewName;
   return $this->idname;
}
function setSize($NewSize) {
   $this->size = $NewSize;
   return $this->size;
}
function setMaxlength($NewMaxlength) {
   $this->maxlength = $NewMaxlength;
   return $this->maxlength;
}
function setValue($NewValue) {
   $this->value = $NewValue;
   return $this->value;
}
function setTabindex($NewTabindex) {
   $this->tabindex = $Tabindex;
   return $this->tabindex;
}
function setMode($NewMode) {
   $this->mode = $NewMode;
   return $this->mode;
}
function displayInputText(){
   print("<input name=\"".$this->idname."\"
         type=\"text\"
         id=\".\this->idname."\"
```





#### 5.2.5.16.8 Das Script: InputElementRadiobutton.php

Im Folgenden sehen Sie das weitgehend unkommentierte Script für die Erstellung von Radiobuttons. Ich werde nur dort Kommentare einbringen, wo sich Erweiterungen gegenüber dem vorherigen Script ergeben haben.

```
<?php

class InputElementRadiobutton {

    // Eigenschaften

    var $idname;

    var $value;

    var $value;

    var $checked;

    // Der Konstruktor

    function InputElementRadiobutton($idname,

        $value,

    $checked="checked\""){
```

Beachten Sie den dritten Parameter \$checked. Hierbei handelt es sich um einen optionalen Parameter. Der Parameter ist mit einem Standardwert belegt, so dass der Konstruktor sowohl mit zwei als auch mit drei Parametern aufgerufen





Web-Programmierung

Dialoge erstellen

5.2.5 Seite 80 werden kann. Wird der Konstruktor mit nur zwei Parametern aufgerufen, so enthält der nicht übergebene dritte Parameter den Standardwert.

```
$this->setIdName($idname);
       $this->setValue($value);
       $this->setChecked($checked);
      }
      // Set-Methoden
      function setIdName($NewName) {
         $this->idname = $NewName;
          return $this->idname;
      }
      function setValue($NewValue) {
          $this->value = $NewValue;
          return $this->value;
      }
      function setChecked($NewChecked) {
          $this->checked = $NewChecked;
          return $this->checked;
      }
      function displayInputRadiobutton(){
          print("<input name=\"$this->idname\"
type=\"radio\"
value=\"$this->value\"
$this->checked />");
          ;
      }
}
?>
```



**Ein ITP Handbuch** 

#### IBM POWER Systems i Softwareentwicklung und -modernisierung

```
<?php
class InputElementButton {
      // Eigenschaften
      var $idname;
      var $type;
      var $value;
       var $event;
      // Der Konstruktor
      function InputElementButton($idname,
                                    $type,
                                    $value,
                                    $event){
           $this->setIdName($idname);
           $this->setType($type);
           $this->setValue($value);
           $this->setEvent($event);
      }
      // Set-Methoden
       function setIdName($NewName) {
          $this->idname = $NewName;
          return $this->idname;
       }
```

#### 5.2.5.16.9 Das Script: InputElementButton.php

Im Folgenden sehen Sie das weitgehend unkommentierte Script für die Erstellung von Druckknöpfen. Ich werde nur dort Kommentare einbringen, wo sich Erweiterungen gegenüber dem vorherigen Script ergeben haben.









Seite 83

#### 5.2.5.17 Stored Procedures einsetzen

Waren unsere bisherigen Optimierungen im Bereich der Benutzerschnittstelle wirksam, rückt der Aspekt der Wiederverwendbarkeit von Datenbankfunktionen in den Vordergrund.

Dabei möchte ich Ihre Aufmerksamkeit auf die Ausführung des SQL-Befehls richten. Mich stört z. B., dass die komplette Befehlsaufbereitung und dann auch die Befehlsausführung im PHP Script realisiert wird. Ändert sich der Zugriff, so muss ich mein Script ändern.

Ich plädiere daher dafür, die Befehlsaufbereitung aus dem Script herauszunehmen und den Zugriff in eine SQL Stored Procedure (SP) zu kapseln.



Kapitel

Web-Programmierung

Dialoge erstellen

# 5.2.5 5.2.5.17.1 Seite 84 Datenbankzugriff

Im i5/OS erstelle ich folgende Stored Procedure GetKunden:

CREATE PROCEDURE GetKunden ( IN SUCH CHAR(60) )
RESULT SETS 1
LANGUAGE SQL
P1: BEGIN
Cursor deklarieren
DECLARE cursor1 CURSOR FOR
SELECT KLIDNR,
KLNAME,
(SELECT SLBEZ
FROM EPSW0402.SVSLST00
WHERE SLKEY = 'LAND' AND
SLWERT = KL.KLLANDKZ)
AS KLLAND,
KLPLZ,
KLORT,
KLSTR,
KLDATN
FROM EPSW0402.SVKLST00 AS KL
WHERE KLIDNR BETWEEN 1000000 AND 6999999 AND
UPPER(KLNAME) LIKE
UPPER('%'    TRIM(SUCH)    '%');
OPEN cursor1;
END P1



Die Stored Procedure erwartet einen Eingabeparameter SUCH, 60, Character.	
In der Prozedur wird der SQL-Befehl, den wir in unserem PHP-Script db2functions.php in die Funktion ExecuteDB2Sql() eingebettet hatten, ausgeführt.	
Die Funktion gibt den kompletten Resultset an den aufrufenden Prozess zurück (RESULT SETS 1).	
Analog hierzu erstelle ich die Stored Procedure GetLieferanten:	
CREATE PROCEDURE GetLieferanten ( IN SUCH CHAR(60) )	
RESULT SETS 1	
LANGUAGE SQL	
P1: BEGIN	
Cursor deklarieren	
DECLARE cursor1 CURSOR FOR	
SELECT KLIDNR,	
KLNAME,	
(SELECT SLBEZ	
FROM EPSW0402.SVSLST00	
WHERE SLKEY = 'LAND' AND	
SLWERT = KL.KLLANDKZ)	
AS KLLAND,	
KLPLZ,	
KLORT,	
KLSTR,	
KLDATN	
FROM EPSW0402.SVKLST00 AS KL	
WHERE KLIDNR >= 7000000 AND	
UPPER(KLNAME) LIKE	
UPPER('%'    TRIM(SUCH)    '%');	
OPEN cursor1;	
END P1	



Kapitel

5.2.5 Seite 85

Web-Programmierung

Dialoge erstellen



5.2.5 Seite 86

#### 5.2.5.17.2 Aufruf der Stored Procedures

Ich kopiere mein Script db2functions.php im Pfad /includes in die Scriptdatei db2functionSP.php und ändere die Funktion ExecuteDB2Sql() wie folgt:

```
function ExecuteDB2Sql($DBHandle,
                          $Selection,
                          $Searchargument) {
      if ($Selection== "KD") {
        $query =
         "call epsw0402.GetKunden('".$Searchargument."')";
      }
      else {
      $query =
      "call epsw0402.GetLieferanten('".$Searchargument."')";
  }
      $ResultSet = db2_exec($DBHandle, $query);
      if (!$ResultSet) {
        die("Status: ". db2_stmt_error(). " " .
                        db2_stmt_errormsg());
      }
      return $ResultSet;
  }
```



Grundsätzlich musste der Gesamtdialog nicht umgestellt werden. Das ist ein Erfolg unserer Optimierungs- und Modularisierungsmaßnahmen beim Datenbankzugriff.

Nur in der Funktion ExecuteDB2Sql() musste der komplexe alte Query-String gegen den relativ einfachen Aufruf der Stored Procedures ausgetauscht werden.

#### 5.2.5.17.3 Programmaufrufe mit Stored Procedures

Das System i kennt das Konstrukt der External Procedures. Diese Prozedurart stellt eine Sonderform der Stored Procedures dar. Mit External Stored Procedures können Programme, die Sie in RPG, Cobol, CL und Java bereits realisiert haben, aufgerufen werden.

#### 5.2.5.17.3.1 Die Aufgabe

In den von uns zuletzt erstellten Dialog soll eine weitere Funktion eingebaut werden, die es möglich macht, die Daten eines Kunden zu drucken.

Diese Funktion war bisher in ein RPG-Programm eingebaut, ist zum Glück aber in ein CL-Programm gekapselt worden.

prtkl: pgm parm(&KLIDNR)

dcl &klidnr \*dec (7 0)
dcl &aklidnr \*char 7

chgvar &aklidnr &klidnr

STRQMQRY QMQRY(EPSW0402P/PRTKL) +
OUTPUT(\*PRINT) NAMING(\*SQL) +
SETVAR((KLIDNR &AKLIDNR))

exit:
endpgm





# Web-Programmierung

Dialoge erstellen

# 5.2.5 Seite 88

Dieses Programm empfängt die Identnummer eines Kunden bzw. Lieferanten. Der Parameter wird an ein SQL-Query übergeben, das dann den entsprechenden Geschäftspartner druckt.

Welche Arbeitsschritte müssen vollzogen werden, damit wir dieses Programm via Stored Procedure aufrufen können?

- 1. In die tabellarische Anzeige der Kunden und Lieferantendaten des Scriptes svklmasterSP.php muss ein Link auf den Aufruf des Programms eingebaut werden.
- 2. Wir müssen eine Funktion entwickeln, die via Stored Procedure das Programm PRTKL aufruft.

#### 5.2.5.17.3.1.1 Grafischer Link

Ich beabsichtige die Druckfunktion im Dialog als grafischen Verweis zu gestalten. Mit folgender HTML-Anweisung lässt sich das realisieren.

```
<a href="printKL.php"><img src="prt.jpg" width="32"
height="32" border="0" alt="drucken"></a>
```

Grundsätzlich besteht hier die Verweisfunktionalität aus zwei Tags:

#### 1. <a href="printKL.php">

Die eigentliche Link-Funktion: Wird auf den Link geklickt, so wird ein PHP-Script PrintKL.php aufgerufen, das den Aufruf der Druckfunktion beinhaltet.

# 2. "><img src="Images/prt.jpg" width="32" height="32" border="0" alt="drucken"></a>

Die Angabe, wo das Drucker-Icon und Formatierungsangaben zu finden sind.



# **E**

#### 5.2.5.17.3.1.2 External Stored Procedure aufrufen

Den oben bezeichneten Link baue ich in die Funktion DisplayDB2ResultSet() in dem Script db2functionsSP.php ein.

```
function DisplayDB2ResultSet($ResultSet) {
   RowCount = 0;
   print("<TABLE width=\"95%\" border=\"0\">");
      print("<TBODY>");
        print("<TR>");
          print("<TH>Funktionen</TH>");
          print("<TH>Identnr.</TH>");
          print("<TH>Name</TH>");
            print("<TH>Land</TH>");
            print("<TH>PLZ</TH>");
            print("<TH>Ort</TH>");
            print("<TH>Strasse</TH>");
            print("<TH>Telefon</TH>");
            print("<TH>Kunde seit</TH>");
          print("</TR>");
   while(db2_fetch_row($ResultSet))
    {
      $RowCount = $RowCount + 1;
      $klidnr
              = db2_result($ResultSet, 'KLIDNR');
      $k1name
               = db2_result($ResultSet, 'KLNAME');
      $klland
               = db2_result($ResultSet, 'KLLAND');
      $klplz
                = db2_result($ResultSet, 'KLPLZ');
```



	Web-Prog	Irammie	rung			
Kal	Dialoge ers	tellen				
5.2.5			\$klort	<pre>= db2_result(\$ResultSet;</pre>	'KLORT');	
Seite 90			\$klstr	<pre>= db2_result(\$ResultSet;</pre>	'KLSTR');	
			\$kltel	<pre>= db2_result(\$ResultSet;</pre>	'KLTEL');	
			\$kldatn	<pre>= db2_result(\$ResultSet;</pre>	'KLDATN');	
			print(" <t< th=""><th>R&gt;");</th><th></th></t<>	R>");		
			print("	<td><a href='\"printKL.php&lt;/th'><th>o?klidnr=\$klidnr\"&gt;</th></a></td>	<a href='\"printKL.php&lt;/th'><th>o?klidnr=\$klidnr\"&gt;</th></a>	o?klidnr=\$klidnr\">
				<img src='\"Images/pri&lt;/th'/> <th>.gif\"</th>	.gif\"	
				width=\"16\"		
				height=\"16\"		
				border= $"1$ "		
				alt=\"drucken\">		
				");		
			print(	<pre>("<td>\$klidnr</td>");</pre>	\$klidnr	
			print(	<pre>("<td>\$k1name</td>");</pre>	\$k1name	
			print(	(" <td>\$k11and</td> ");	\$k11and	
			print(	<pre>("<td>\$klplz </td>");</pre>	\$klplz	
			print(	<pre>("<td>\$klort </td>");</pre>	\$klort	
			print(	(" <td>\$klstr </td> ");	\$klstr	
			print(	<pre>("<td>\$kltel </td>");</pre>	\$kltel	
			print(	<pre>("<td>\$kldatn</td>");</pre>	\$kldatn	
			print	("		
		}				
		I	print(" <th>BODY&gt;");</th> <th></th>	BODY>");		
		pr	int(" <th>BLE&gt;");</th> <th></th>	BLE>");		





5.2.5 Seite 91

```
return $RowCount;
```

}

Achten Sie auf die fett-markierten Zeilen:

- 1. Ich habe der Tabelle eine neue Spaltenüberschrift hinzugefügt.
- 2. Wichtig ist das  $\langle a \rangle$  Tag:

<a href=\"printKL.php?klidnr=\$klidnr\">

An den Aufruf des Scripts printKL.php hänge ich die Parameterübergabe als Name-Value-Paar an.

3. Den Parameter empfange ich dann mit der GET-Methode im Script printKL. php.

```
<?php

require("includes/db2functionsSP.php");

$connectionstring = ConnectToDB2($system,

$user,

$passwd);

$Rc = ExecuteDB2Pgm($connectionstring, $_GET['klidnr']);

if (RC) {

print("Der Kundensatz wurde gedruckt.");

}

CloseDB2Connection($connectionstring);

?>
```



apite	Web-Programmierung			
Ŷ	Dialoge erstellen			
5.2.5 Seite 92	Sie sehen das Empfangen des Parameters in der Parameterleiste des Funktions- aufrufs ExecuteDB2Pgm.			
	Im Folgenden sehen Sie den Aufruf der Stored Procedure ExecuteDB2Pgm() im Script db2functionsSP.php:			
	<pre>function ExecuteDB2Pgm(\$DBHandle, \$klidnr) {</pre>			
	<pre>\$query = "call epsw0402.prtkl(".\$klidnr.")";</pre>			
	<pre>\$Rc = db2_exec(\$DBHandle, \$query);</pre>			
	if (!\$Rc) {			
	die("Status: ". db2_stmt_error(). " " .			
	<pre>db2_stmt_errormsg());</pre>			
	}			

}

Sie sehen, dass sich der Aufruf einer External Stored Procedure nicht von dem einer SQL Stored Procedure unterscheidet.



#### 5.2.5.17.3.1.3 Die External Stored Procedure

5.2.5 Seite 93

_	
	CREATE PROCEDURE EPSW0402.PRTKL(IN KLIDNR DECIMAL (7 , 0))
	LANGUAGE CL
	NOT DETERMINISTIC
	NO SQL
	EXTERNAL NAME EPSW0402P.PRTKL
	PARAMETER
	STYLE GENERAL

#### Vorteil:

Rufen wir System i Programme über External SPs auf, so handelt es sich, von der Anwendungsseite aus betrachtet, um das Nutzen einer Standard-Schnittstelle. Damit ist es durchaus denkbar, dass ich weiterhin mit den DB2-Funktionen des PHP weiterarbeiten kann, wenn sich die Umgebung der PHP-Anwendung ändert, es muss lediglich die SP auf dem Server ausgetauscht werden.

#### Nachteil:

Die Parameter, die ich an die SP übergebe, sind de facto Input-Parameter, auch wenn ich Sie als INOUT bzw. OUT deklariert habe. Damit schränkt sich der Einsatz der External SP auf solche Fälle ein, bei denen wir zwar Parameter an eine Serverfunktion übergeben, jedoch nichts zurückerwarten.

Um diese Problemzone könnten wir "drumherumprogrammieren", dieses ist aus meiner Sicht heraus jedoch nicht unbedingt notwendig, denn Zendcore stellt für uns System i Anwender Funktionen bereit, die es uns möglich machen optimal mit System i Objekten zu kommunizieren: Das **i5 PHP API Toolkit.** 

## 5.2.5 Seite 94

#### 5.2.5.18 Zend API für i5/OS

Neben der Möglichkeit mit Standard PHP-Funktionen i5/OS-Datenbanken anzusprechen, hält Zendcore für i5/OS noch eine Reihe spezifischer Funktionen für das Betriebssystem i5/OS bereit. Diese Funktionen werden zusammengefasst unter dem Begriff i5 PHP API Toolkit.

Ich werde hier nicht alle Funktionen mit Ihnen besprechen, sondern nur die, die sich dem Anwendungsentwickler kaufmännischer Software geradezu aufdrängen.

- DB2 für System i spezifische Funktionen
- i5/OS Programmaufrufe
- i5/OS Befehlsaufrufe
- i5/OS Programme als Webservices ansprechen

#### 5.2.5.18.1 i5/OS Programmaufrufe

Eine der zentralen Funktionen des i5 PHP API Toolkits besteht zumindest aus der Sicht des Anwendungsentwicklers kaufmännischer Anwendungen darin, native Programme des i5/OS aufrufen zu können.

Auch hier steht der Aspekt der Wiederverwendbarkeit im Vordergrund. Wäre es nicht schön, wenn wir z. B. ein bestehendes Druckprogramm, das Parameter-gesteuert aufgerufen wird, aus einer Webanwendung heraus aufrufen könnten?

Sicherlich können wir via DB2-SQL-Funktionen auf die Datenbank zugreifen. Wenn wir jedoch feststellen, dass unsere Dateien und Tabellen sich beim Zugriff mit SQL nicht optimal verhalten, so könnten wir ja auch auf den Gedanken kommen mit nativen RPG- bzw. Cobol-Zugriffsmodulen Daten anzusprechen. Manchmal ist ein gezieltes CHAIN bzw. READ einfach schneller als ein SQL Select.

Die Vorstellung, dass ich unter i5/OS zentrale Datenbankzugriffmodule erzeuge, die von unterschiedlichen Applikationen verwendet werden können – von PHP-, von Java-, von RPG-Applikationen, liegt sicherlich in nicht allzu weiter Ferne.

Genau an dieser Stelle möchte ich ansetzen.

Wir werden gemeinsam das bestehende Script svmastersp.php, das Kundenund Lieferantendaten enthält, so modifizieren, dass wir daraus in eine Einzelsatzverarbeitung übergehen können.





#### 5.2.5.18.1.1 Die Aufgabe

Wir werden ein Script realisieren, mit dem Kunden/Lieferanten neu erfasst, geändert und gelöscht werden können.

Zu diesem Zweck füge ich in die Tabellenanzeige der Kunden-/Lieferantendaten folgende Zeilen in das Script db2FunctionsSP.php in der Funktion DisplayDB2ResultSet() ein.

function DisplayDB2ResultSet(\$ResultSet) {	
<pre>\$RowCount = 0;</pre>	
<pre>print("<table border='\"0\"' width='\"95%\"'>");     print("<tbody>");     print("<tboty>");     print("<tb>Punktionen");     print("<tb>Name");     print("<tb>Name");     print("<tb>Name");     print("<tb>Name");     print("<tb>Name");     print("<tb>TBOTK/TB&gt;");     print("<tb>TBOTK/TB&gt;");     print("<tb>TBOTK/TB&gt;");     print("<tb>TBOTK/TB&gt;");     print("<tb>TBOTK/TB&gt;");     print("<tb>Kasse");     print("<tb>Kasse</tb>");     print("<tb>Kasse</tb></tb></tb></tb></tb></tb></tb></tb></tb></tb></tb></tb></tb></tboty></tbody></table></pre>	
<pre>\$RowCount = \$RowCount + 1;</pre>	
<pre>\$klidnr = db2_result(\$ResultSet, 'KLIDNR'); \$klname = db2_result(\$ResultSet, 'KLIANE'); \$klland = db2_result(\$ResultSet, 'KLIAND'); \$klplz = db2_result(\$ResultSet, 'KLPLZ'); \$klort = db2_result(\$ResultSet, 'KLORT'); \$klstr = db2_result(\$ResultSet, 'KLSTR'); \$kltel = db2_result(\$ResultSet, 'KLTEL'); \$kldatn = db2_result(\$ResultSet, 'KLDATN'); \$kldatn = db2_result(\$ResultSet, 'KLDATN'); </pre>	
<pre>print("<tr>");</tr></pre>	
<pre>print("<td><a href="\&quot;svklDetailPC.php?klidnr=\$klidnr&amp;mode=UPDATE&amp;type=&quot;.\$_POST['GPTyp'].&quot;\&quot;"></a></td></pre>	<a href="\&quot;svklDetailPC.php?klidnr=\$klidnr&amp;mode=UPDATE&amp;type=&quot;.\$_POST['GPTyp'].&quot;\&quot;"></a>
<pre>print("<a href="\&quot;svklDetailPC.php?klidnr=\$klidnr&amp;mode=DELETE&amp;type=&quot;.\$_POST['GPTyp'].&quot;\&quot;"></a></pre>	
<pre><img alt='\"ändern\"' border='\"1\"' height='\"16\"' src='\"Images/loeschen.gif\"' width='\"16\"'/> ");</pre>	

Grafik Links einbauen

In die Tabellenanzeige füge ich zwei grafische Links ein, die folgende Parameter an das Script svDetail.php über die Methode GET übertragen.

Klidnr: Die Identnummer des angewählten Satzes. Sie ist nötig, um später den gewünschten Satz aus der Datei SVKLST00 auszulesen, dann zu ändern oder zu löschen.

Mode: Mit Hilfe der Parameterwerte UPDATE bzw. DELETE wird in der Einzelsatzverarbeitung gesteuert, ob der angewählte Satz aktualisiert oder gelöscht wird.



5.2.5 Seite 95 Web-Programmierung

Dialoge erstellen

# 5.2.5 Seite 96

Kunden u	nd Lieferant	en suchen	
Suchbegrift	f	ds	
Kunden / L	ieferanten	00	
		Suchen Initialisieren	
Funktione	en Identnr. N	lame	Land
S 8	1014500 K	raft Foods Deutschland GmbH & Co. KG	Deutschland
	1027400 ⊢	larrods Knightsbridge Limited	Groß Britannien
🖾 🔕 📴	1042000 B	erendsohn AG	Deutschland
🖾 🔕 🔯	1064000 J	D Edwards Deutschland GmbH	Deutschland
🖾 🔕 🐷	1089900 V	/ort+Bild Verlag Konradshöhe GmbH & Co.	Deutschland
S 🛛 🖸	1142600 A	DS Informatik Hard- u. Software-Beratung	Deutschland
🔍 🔕 🔛	1202600 S	mith + Nephew Richards GmbH Medizintechnik	Deutschland
🔍 🔕 🔛	1205800 D	IFA Deutsche Immobilien Fonds AG	Deutschland

Tabellarische Anzeige

Type: Der Wert dieses Parameters wird aus dem Suchdialog heraus versorgt und übergibt die Werte KD bzw. LF, so dass unterschieden werden kann, ob es sich beim angewählten Satz um einen Kunden oder Lieferanten handelt. Diese Unterscheidung ist wichtig bei der Anzeige einzelner Felder im Einzelverarbeitungsdialog.

Die Einzelsatzverarbeitung wird realisiert mit Hilfe folgender Dokumente, die strukturell auf das aufbauen, was wir uns bisher erarbeitet haben:

svkldetailPC.php: Das Rahmendokument, das uns dabei hilft, den Dialog in einzelne Komponenten zu strukturieren.

svklPreDetail.php: Die der Dialoganzeige vorangehende Anfangsverarbeitung. Übergebe ich im Parametermode die Werte UPDATE oder DELETE, so muss ein vorhandener Datensatz aus der Datei SVKLST00 gelesen werden. Soll ein neuer Satz erfasst werden (mode=NEW), dann bedarf es keiner Vorverarbeitung. Ein leerer Dialog muss angezeigt werden.

svkldisplayPC.php: In diesem Script wird der eigentliche Dialog angezeigt.

svklPostDetail.php: Dieses Script ändert, löscht oder erfasst die Kunden-/ Lieferantendaten.

Kunde ändern	
Firma / Identnummer / Unsere Kundenummer	1 1014500
Anrede / Name	Firma 🔄 Kraft Foods Deutschland GmbH & Co. KG
Strasse	Langemarckstr. 4-20
Postfach / Postfach PLZ / Postfach Ort	
Land / Postleitzahl / Ort	Deutschland 🗾 28199 Bremen
Telefon / Kurzwahl	0421/59901 e
Fax / Kurzwahl	0421/5994539
Handy / Kurzwahl	
Internet / E-Mail(allgemein)	www.kraft.de test@bremen.de
Bank / Bankleitzahl / Konto	
Datum Anlage / Benutzer	2002-12-03 HORVAT
Datum Änderung / Benutzer	2002-12-03 KOSBAHN
	speichern Initialisieren

#### Einzelsatzverarbeitung

In der Folge werden wir noch weitere Scripte zur besseren Strukturierung erzeugen, die jedoch derzeit keine Rolle spielen.





Seite 97

#### 5.2.5.18.1.2 Das Script svkldetailPC.php

Sehen Sie sich bitte den Inhalt dieses Scripts an.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
          charset=iso-8859-1" />
    <LINK href="Styles/format.css" rel="stylesheet"
          type="text/css">
    <script src="js/formhandling.js"</pre>
            type="text/javascript"></script>
    <title>Kunden und Lieferantendetails</title>
  </head>
  <?php
  require("includes/localSettings.php");
 if ($_GET['mode'] == "UPDATE" OR
      $_GET['mode'] == "DELETE") {
     require("svklPreDetail.php");
     }
```





Web-Programmierung

Dialoge erstellen

5.2.5 Seite 98 In der oben dargestellten Abfrage wird über die Variable \$\_GET[,mode'] abgefragt, ob ein Datensatz gelöscht oder geändert werden soll. In diesem Fall wird das Vorverarbeitungsscript svklPreDetail.php mittels require() angezogen.

```
print("<body onload=\"javascript:return
pageHandlerKL();\">");
```

require("svkldisplayPC.php");

An dieser Stelle wird mittels require() der eigentliche Bearbeitungsdialog angezogen.

	<pre>print("");</pre>
	?>
<	/html>



Ergänzung 10/2009

Dialoge erstellen

#### 5.2.5.18.1.3 Das Script svklPreDetail.php

In diesem Script wird das SQLRPG-Programm GETCUSTSQL aufgerufen. Sehen Sie sich den Quelltext an:

d	klfinr	s		2p	0		Ī
d	klidnr	s		7p	0		
d	klankz	s		2p	0		
d	klanrede	s		30A			
d	klname	s		100A			
d	klstr	s		32A			
d	klpstf	s		10A			
d	klplzp	s		37A			
d	klortp	s		25A			
d	kllandkz	s		3A			
d	klland	s		30A			
d	klplz	s		7A			
d	klort	s		25A			
d	kltelk	s		3A			
d	kltel	s		20A			
d	klfaxk	s		3A			
d	klfax	s		20A			
d	klftfk	s		3A			
d	klftf	s		20A			
d	klinter	s		60A			
d	kleml	s		60A			
d	klulfnr	s		20A			
d	klbank	s		30A			
d	klblz	s		8A			
d	klkto	s		15A			
d	kldatn	s		10A			
d	klusrn	s		10A			
d	kldatc	s		10A			
d	klusrc	s		10A			
с	*entry		plist				
С			parm			klfinr	
С			parm			klidnr	
С			parm			klankz	
С			parm			klanrede	
С			parm			klname	



5.2.5

Seite 99

Kapitel

Web-Programmierung
Dialoge erstellen

5.2.5 Seite 100

с	parm	klstr
с	parm	klpstf
с	parm	klplzp
С	parm	klortp
С	parm	kllandkz
С	parm	klland
С	parm	klplz
С	parm	klort
С	parm	kltelk
С	parm	kltel
С	parm	klfaxk
С	parm	klfax
С	parm	klftfk
С	parm	klftf
С	parm	klinter
С	parm	kleml
С	parm	klulfnr
С	parm	klbank
С	parm	klblz
С	parm	klkto
С	parm	kldatn
С	parm	klusrn
С	parm	kldatc
С	parm	klusrc
/free		
exec sql select	klfinr,	
	klidnr,	
	klankz,	
	(select slbez	
	from epsw0402.svslst	:00
	where $slwert = kl.kl$	ankz AND
	slkey = 'ANKZ'	) as
	KLANREDE,	
	KLNAME,	
	KLSTR,	

Web-Programmierung Dialoge erstellen



5.2.5

Seite 101

	KLPSTF,
	KLPLZP,
	KLORTP,
	KLLANDKZ,
	(select slbez
	from epsw0402.svslst00
	where slwert = kl.kllandkz AND
	slkey = 'LAND') as
	KLLAND,
	KLPLZ,
	KLORT,
	KLTELK,
	KLTEL,
	KLFAXK,
	KLFAX,
	KLFTFK,
	KLFTF,
	KLINTER,
	KLEML,
	KLULFNR,
	KLBANK,
	KLBLZ,
	KLKTO,
	char(KLDATN, ISO) as KLDATN,
	KLUSRN,
	char(KLDATC, ISO) as KLDATC,
	KLUSRC
into	:klfinr,
	:klidnr,
	:klankz,
	:klanrede,
	:klname,
	:klstr,
	:klpstf,
	:klplzp,
	:klortp,





bitel	Web-Programmierung					
Ka	Dialoge erst	ellen				
5.2.5		:	kllandkz,			
Seite 102		:	klland,			
		:	klplz,			
		:	klort,			
		:	kltelk,			
		:	kltel,			
		:	klfaxk,			
		:	klfax,			
		:	klftfk,			
		:	klftf,			
		:	klinter,			
		:	kleml,			
		:	klulfnr,			
		:	klbank,			
		:	klblz,			
		:	klkto,			
		:	kldatn,			
		:	klusrn,			
		:	kldatc,			
		:	klusrc			
		from ep	sw0402.svklst00 kl			
		where K	LIDNR = :klidnr;			
		<pre>*inlr = *on;</pre>				
		/end-free				

Dieses Programm ist von seiner Aufgabenstellung als einfach anzusehen. Ich habe darüber hinaus auch sämtliche Seitenlogiken, wie Fehlerabfragen etc. entfernt, um nicht den Blick auf das Wesentliche dieses Programms zu verstellen.

In diesem Programm werden Parameter definiert und mittels \*ENTRY PLIST empfangen und zurückgegeben.

Das Einlesen der Daten geschieht mit Hilfe des SQL-Befehls Select.

Wird das Programm GESTCUSTSQL von einem PHP-Script aufgerufen, so wird die Identnummer (klidnr) an das RPG-Programm übergeben. Dann wird der entsprechende Satz gelesen und an das PHP-Script zurückgegeben.



Web-Programmierung Dialoge erstellen Um ein Programm aufzurufen benötigen wir vier Funktionen des i5 API Toolkits: Seite 103 • i5\_connect

- i5\_program\_prepare
- i5\_program\_call
- i5\_close

#### 5.2.5.18.1.3.1 **Die Funktion i5\_connect**

Die Funktion i5\_connect eröffnet eine Verbindung zum i5/OS.

Sehen Sie sich bitte die Syntax an:

#### resource i5\_connect(

string	server,
string	user,
string	password
[, arra	ay options]).

Die Funktion erwartet mindestens drei Parameter:

Server: Hostname oder IP-Adresse des System i

User: Benutzerprofilname

Password: Kennwort

Die Funktion gibt ein Verbindungshandle zurück.



# 5.2.55.2.5.18.1.3.2Seite 104Die Funktion i5\_program\_prepare

Die Funktion i5\_program\_prepare dient dazu, dem PHP-Umfeld die Parameterübergabe bekanntzumachen.

Syntax:

resource i5_program_prepare(
string name
[, array description]
[, resource connection]).

Die Funktion erwartet mindestens einen Parameter.

Name: Aufrufname und Pfad des System i Programms, das aufgerufen werden soll.

Description: Die Parameterleiste, die an das Programm übergeben werden soll. Erwartet das aufzurufende Programm keine Parameter, so wird dieser Funktionsparameter weggelassen.

Connection: Das Verbindungshandle, das von i5\_connect() bereitgestellt wird.

Die Funktion gibt ein Programmhandle zurück.

#### 5.2.5.18.1.3.3 Die Funktion i5\_program\_call

Diese Funktion realisiert den Aufruf des System i Programms.

pool	i5_program_call(		
	resource program,		
	array params		
	[, array retvals]).		

Parameter:

Programm: Das von i5\_programm\_prepare() ermittelte Programmhandle

Params: Ein Array, das die Werte der Input-Parameter übergibt.



#### Achtung!

Arbeiten Sie mit Standard RPG-Programmen, so sind alle Parameter per Definitionen INOUT-Parameter. Ergo müssen, um Datenkonflikte zu vermeiden, auch alle Parameter mit einem sinnvollen Wert vorbelegt werden.

Retvals: Gibt unser Programm Werte zurück, so werden durch dieses Array die Felder, die die Rückgabewerte enthalten, im PHP-Script definiert.

Auch diese Funktion gibt ein Handle zurück, das kennzeichnet, ob der Programmaufruf erfolgreich oder nicht erfolgreich (false) durchgeführt wurde.

#### 5.2.5.18.1.3.4 Die Funktion i5\_close

Diese Funktion schließt die Verbindung zum System i.

```
bool i5_close([resource connection]).
```

Sie erwartet den optionalen Parameter connection. Dieses ist der Verbindungshandle, den i5\_connect() ermittelt hat.

Sehen wir uns jetzt das Script svklPreDetail.php an:

```
<?php

<connectionstring = i5_connect( $host,

$user,

$passwd);
```

Hier sehen Sie den Aufruf der Funktion i5\_connect(). Die Parameterwerte werden durch localSettings.php gesetzt.

```
if (!$connectionstring) {
   die("Fehler in Verbindungsaufnahme (connect)".
        i5_errormsg());
}
```

Etwaige Verbindungsfehler werden abgefangen. Mit der Funktions i5\_ errormsg() wird der Nachrichtentext für eine gescheiterte Verbindungsaufnahme angezeigt. Konnte die Verbindung nicht hergestellt werden, so endet das Script hier.

```
require(,,includes/DSCustomer.php");
```







# Web-Programmierung

**Dialoge erstellen** 

5.2.5 Seite 106 Mittels require() wird die Parameterbeschreibung für die Funktion i5\_programm\_prepare() inkludiert. Möglicherweise brauchen wir diese Parameterbeschreibung auch noch in anderen Scripten.

Hier sehen Sie den Inhalt von DSCustomer.php:

```
$customer =array(
        "klfinr" => array("name"=>"KLFINR",
                            "type" => I5_TYPE_PACKED,
                            "io"=>I5_INOUT,
                            "length" => "2.0"),
         "klidnr" => array("name"=>"KLIDNR",
                            "type" => I5_TYPE_PACKED,
                            "io"=>I5_INOUT,
                            "length" => "7.0"),
        "klankz" => array("name"=>"KLANKZ",
                            "type" => I5_TYPE_PACKED,
                            "io"=>I5_INOUT,
                            "length" => "2.0"),
        "klanrede"=> array("name"=>"KLANREDE",
                            "type" => I5_TYPE_CHAR,
                            "io"=>I5_INOUT,
                            "length" => "30"),
        "klname" => array("name"=>"KLNAME",
                            "type" => I5_TYPE_CHAR,
                            "io"=>I5_INOUT,
                            "length" => "100"),
        "klstr" => array("name"=>"KLSTR",
                            "type" => I5_TYPE_CHAR,
                            "io"=>I5_INOUT,
                            "length" => "32"),
         "klpstf" => array("name"=>"KLPSTF",
                            "type" => I5_TYPE_CHAR,
                            "io"=>I5_INOUT,
                            "length" => "10"),
         "klplzp" => array("name"=>"KLPLZP",
                            "type" => I5_TYPE_CHAR,
                            "io"=>I5_INOUT,
                            "length" => "7"),
        "klortp" => array("name"=>"KLORTP",
                            "type" => I5_TYPE_CHAR,
                            "io"=>I5_INOUT,
                            "length" => "25"),
```




Seite 107

"kllandkz"=> array("name"=>"KLLANKZ", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "3"), "klland" => array("name"=>"KLLAND", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "30"), "klplz" => array("name"=>"KLPLZ", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "7"), "klort" => array("name"=>"KLORT", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "25"), "kltelk" => array("name"=>"KLTELK", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "3"), "kltel" => array("name"=>"KLTEL", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "20"), "klfaxk" => array("name"=>"KLFAXK", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "3"), "klfax" => array("name"=>"KLFAX", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "20"), "klftfk" => array("name"=>"KLFTFK", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "3"), "klftf" => array("name"=>"KLFTF",





**Dialoge erstellen** 

5.2.5

**Seite 108** 



"klulfnr" => array("name"=>"KLULFNR", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "20"), "klbank" => array("name"=>"KLBANK", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "30"), "klblz" => array("name"=>"KLBLZ", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "8"), "klkto" => array("name"=>"KLKTO", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "15"), "kldatn" => array("name"=>"KLDATN", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "10"), "klusrn" => array("name"=>"KLUSRN", "type" => I5\_TYPE\_CHAR, "io"=>I5\_INOUT, "length" => "10"), "kldatc" => array("name"=>"KLDATC",



Kapitel

5.2.5

Seite 109

```
"io"=>I5_INOUT,
"length" => "10"),
"klusrc" => array("name"=>"KLUSRC",
"type" => I5_TYPE_CHAR,
"io"=>I5_INOUT,
"length" => "10")
```

Sie sehen, dass für unser Programm die Parameterleiste doch eine nicht unerhebliche Länge hat. Abgesehen von diesem Umstand lässt sich leicht erkennen, dass die Parameter insgesamt als Array deklariert werden, das aus einzelnen Subarrays besteht, die wiederum die Einzeldeklaration eines Parameters beinhalten. Wir erkennen weiterhin, dass hier grundsätzlich mit assoziativen Arrays gearbeitet wird.

Um einen einzelnen Parameter zu deklarieren, benötigen wir vier Angaben:

1. Name des Parameters: name=

Der Name des Parameters wird in folgender Form angegeben:

Name="Feldname"

);

Der Feldname muss nicht unbedingt mit dem Parameternamen im RPG-Programm übereinstimmen.

2. Typ des Parameters: type=

Der Typ eines Parameters wird durch besondere Einsetzwerte deklariert, die sich selbst beschreiben:

I5_TYPE_CHAR	
I5_TYPE_INT	
I5_TYPE_PACKED	
I5_TYPE_ZONED	
I5_TYPE_FLOAT	
I5_TYPE_BYTE	
I5 TYPE STRUCT	





**Dialoge erstellen** 

5.2.5 Seite 110 Wirkung des Parameters: io=

Hier wird bestimmt, ob es sich bei den Parametern um Input, Out oder Input-Output-Parameter handelt. Auch hier wird mit einer speziellen Syntax gearbeitet.

I5_IN	
I5_OUT	
I5_INOUT	

Länge des Parameters: length=

Die Länge eines Parameters wird numerisch oder als Zeichenfolge deklariert.

Length = 10 : Feld von zehn Byte Länge

Length = "11.2" : numerisches Feld mit elf Stellen – neun Vorkommastellen und zwei Nachkommastellen

Jetzt sind wir zurück im Script svklPreDetail:

<pre>\$proghandle = i5_program_prepare("EPSW0402P/GETCSTSQL",</pre>
<pre>\$customer);</pre>
<pre>if (\$prog === FALSE){</pre>
die("Programmvorbereitung gescheitert
<pre>(prepare)".i5_errormsg());</pre>
}

Hier sehen Sie den Aufruf der Funktion i5\_program\_prepare. Beachten Sie die Variable \$customer, die die Parameterdeklaration enthält.

Sollte die Programmvorbereitung scheitern, so wird wiederum eine Meldung ausgegeben.



Dialoge erstellen



\$inParms =arrav(	5.2.5
"klfinr" => 1	Seite 111
"klidnr" => \$ GFT['klidnr'].	
"klankz" => 1.	
"klanrede"=> " ".	
"klname" => " ".	
"klstr" => " ".	
"klpstf" => " ",	
"klplzp" => " ",	
"klortp" => " ",	
"kllandkz"=> " ",	
"klland" => " ",	
"klplz" => " ",	
"klort" => " ",	
"kltelk" => " ",	
"kltel" => " ",	
"klfaxk" => " ",	
"klfax" => " ",	
"klftfk" => " ",	
"klftf" => " ",	
"klinter" => " ",	
"kleml" => " ",	
"klulfnr" => " ",	
"klbank" => " ",	
"klblz" => " ",	
"klkto" => " ",	
"kldatn" => " ",	
"klusrn" => " ",	
"kldatc" => " ",	
"klusrc" => " "	
);	

Hier sehen Sie jetzt die Werte-Versorgung der Inputparameter. Achten Sie darauf, dass Sie nicht gegen die Parameterkonventionen des System i Programms verstoßen.



5.2.5 Seite 112

<pre>\$outParms</pre>	=arrav(
Jou ci ui iii j	-urruy (

1 1113	-array (	
	"klfinr" =>	"pklfinr",
	"klidnr" =>	"pklidnr",
	"klankz" =>	"pklankz",
	"klanrede"=>	"pklanrede",
	"klname" =>	"pklname",
	"klstr" =>	"pklstr",
	"klpstf" =>	"pklpstf",
	"klplzp" =>	"pklplzp",
	"klortp" =>	"pklortp",
	"kllandkz"=>	"pklandkz",
	"klland" =>	"pklland",
	"klplz" =>	"pklplz",
	"klort" =>	"pklort",
	"kltelk" =>	"pkltelk",
	"kltel" =>	"pkltel",
	"klfaxk" =>	"pklfaxk",
	"klfax" =>	"pklfax",
	"klftfk" =>	"pklftfk",
	"klftf" =>	"pklftf",
	"klinter" =>	"pklinter",
	"kleml" =>	"pkleml",
	"klulfnr" =>	"pklulfnr",
	"klbank" =>	"pklbank",
	"klblz" =>	"pklblz",
	"klkto" =>	"pklkto",
	"kldatn" =>	"pkldatn",
	"klusrn" =>	"pklusrn",
	"kldatc" =>	"pkldatc",
	"klusrc" =>	"pk]usrc"

Hier sehen Sie jetzt das Array, das die Ausgabeparameter im PHP deklariert. Konzentrieren Sie sich einmal nur auf den Parameter klidnr. Der Wert im Programmparameter klidnr wird an die PHP-Variable pklidnr zurückgegeben, die damit als definiert gilt.



**E** 

Seite 113

```
//$outParms = array($outStruct);
$rc = i5_program_call($proghandle, $inParms, $outParms);
if ($rc === FALSE){
    die("Fehler im Programmaufruf (call)".
        i5_errormsg());
}
```

Jetzt erfolgt der Aufruf des Programms mittels der Funktion i5\_program\_ call(). Beachten Sie die Parameterübergabe mittels der Arrays \$inparms und \$outparms.





5.2.5 Seite 114

<pre>\$_POST['klfinr']</pre>	=	<pre>\$pklfinr;</pre>
<pre>\$_POST['klidnr']</pre>	=	<pre>\$pklidnr;</pre>
<pre>\$_POST['klankz']</pre>	=	<pre>\$pklankz;</pre>
<pre>\$_POST['klanrede']</pre>	=	<pre>\$pklanrede;</pre>
<pre>\$_POST['klname']</pre>	=	<pre>\$pklname;</pre>
<pre>\$_POST['klstr']</pre>	=	<pre>\$pklstr;</pre>
<pre>\$_POST['klpstf']</pre>	=	<pre>\$pklpstf;</pre>
<pre>\$_POST['klplzp']</pre>	=	<pre>\$pklplzp;</pre>
<pre>\$_POST['klortp']</pre>	=	<pre>\$pklortp;</pre>
<pre>\$_POST['kllandkz']</pre>	] =	<pre>\$pklandkz;</pre>
<pre>\$_POST['klland']</pre>	=	<pre>\$pklland;</pre>
<pre>\$_POST['klplz']</pre>	=	<pre>\$pklplz;</pre>
<pre>\$_POST['klort']</pre>	=	<pre>\$pklort;</pre>
<pre>\$_POST['kltelk']</pre>	=	<pre>\$pkltelk;</pre>
<pre>\$_POST['kltel']</pre>	=	<pre>\$pkltel;</pre>
<pre>\$_POST['klfaxk']</pre>	=	<pre>\$pklfaxk;</pre>
<pre>\$_POST['klfax']</pre>	=	<pre>\$pklfax;</pre>
<pre>\$_POST['klftfk']</pre>	=	<pre>\$pklftfk;</pre>
<pre>\$_POST['klftf']</pre>	=	<pre>\$pklftf;</pre>
<pre>\$_POST['klinter']</pre>	=	<pre>\$pklinter;</pre>
<pre>\$_POST['kleml']</pre>	=	<pre>\$pkleml;</pre>
<pre>\$_POST['klulfnr']</pre>	=	<pre>\$pklulfnr;</pre>
<pre>\$_POST['klbank']</pre>	=	<pre>\$pk1bank;</pre>
<pre>\$_POST['klblz']</pre>	=	<pre>\$pklblz;</pre>
<pre>\$_POST['klkto']</pre>	=	<pre>\$pklkto;</pre>
<pre>\$_POST['kldatn']</pre>	=	<pre>\$pkldatn;</pre>
<pre>\$_POST['klusrn']</pre>	=	<pre>\$pklusrn;</pre>
<pre>\$_POST['kldatc']</pre>	=	<pre>\$pkldatc;</pre>
<pre>\$_POST['klusrc']</pre>	=	<pre>\$pklusrc;</pre>

Jetzt werden die Parameterwerte an die Felder des Dialogs übertragen.

```
$rc = i5_close($connectionstring);
print("</body>");
if ($rc === FALSE){
    die("Fehler beim Beendigen der Verbindung (close)".
i5_errormsg());
```

Mit Hilfe der Funktion i5\_close() wird die Verbindung zum System i beendet.

Natürlich bestehen an dieser Stelle Optimierungsmöglichkeiten.

Eine erste Maßnahme besteht darin, dass ich die Parameterarrays \$inparms und \$outparms in eigene Quellen stelle, die mittels require() eingebunden werden. Es könnte ja durchaus möglich sein, dass noch weitere PHP-Scripte das RPG GETCUSTSQL aufrufen werden, so dass eine zentrale Speicherung der Parameterleisten als sinnvoll betrachtet werden kann.

## 5.2.5.18.1.3.5 Mit PCML arbeiten

Eine weitere Möglichkeit der Optimierung besteht darin, die Parameterdeklaration für die Funktion i5\_program\_prepare() komplett extern zu betreiben.

Hiermit meine ich nicht, dass dieses Array in eine externe Quellendatei hineingestellt und mittels require() eingebunden wird. Ich meine vielmehr, dass wir die Parameterdeklaration generieren lassen können!

Dazu müssen wir uns ein wenig mit der Programm Communication Language (PCML) beschäftigen. PCML ist ein XML-Derivat und dafür angelegt worden, Programmschnittstellen zu beschreiben.

Keine Angst, wir müssen uns jetzt nicht auch noch mit den Feinheiten der PCML Syntax beschäftigen. Gerade für System i Programmierer ist die Arbeit mit PCML so gut wie kein Problem. Alle ILE Kompilierbefehle haben mittlerweile einen Parameter, der es möglich macht, PCML bei der Programmumwandlung erstellen zu lassen.

Als Beispiel rufe ich einfach den Befehl CRTBNDRPG auf. In den zusätzlichen Parametern sehen Sie folgende Parameter:

RPG-Binderpro	ogramm erstellen	(CRTBNDRPG)	
Auswahl eingeben und Eingabetast	te drücken.		
Verzeichnis einschließen	INCDIR	<u>×</u> NONE	
+ für weitere Werte			
ProgSchnittstelle generieren Datenstromdatei f. ProgSchn.	PGMINFO INFOSTMF	<u>*N0</u>	

# PCML generieren

In den Parameter PGMINFO trage ich den Wert \*PCML ein und in den Parameter INFOSTMF den Pfad für die zu generierende PCML-Datei: /HOME/ GETCSTSQL.PCML

Nach der erfolgreichen Umwandlung des Programms kopiere ich die erzeugte PCML-Datei in den Pfad /pcml meines Projektes.







Hier sehen Sie die erstellte PCML-Datei: <pcml version="4.0"> <!-- RPG program: GETCSTPCML --> <!-- created: 2007-11-08-14.09.25 --> <!-- source: EPSW0402P/QRPGLESRC(GETCSTSQL) --> <program name="GETCSTSQL"</pre> path="/QSYS.LIB/EPSW0402P.LIB/GETCSTSQL.PGM"> <data name="KLFINR" type="packed" length="2"</pre> precision="0" usage="inputoutput" /> <data name="KLIDNR" type="packed" length="7"</pre> precision="0" usage="inputoutput" /> <data name="KLANKZ" type="packed" length="2"</pre> precision="0" usage="inputoutput" /> <data name="KLANREDE" type="char" length="30"</pre> usage="inputoutput" /> <data name="KLNAME" type="char" length="100"</pre> usage="inputoutput" /> <data name="KLSTR" type="char" length="32"</pre> usage="inputoutput" /> <data name="KLPSTF" type="char" length="10"</pre> usage="inputoutput" /> <data name="KLPLZP" type="char" length="37"</pre> usage="inputoutput" /> <data name="KLORTP" type="char" length="25"</pre> usage="inputoutput" /> <data name="KLLANDKZ" type="char" length="3"</pre> usage="inputoutput" /> <data name="KLLAND" type="char" length="30"</pre> usage="inputoutput" /> <data name="KLPLZ" type="char" length="7"</pre> usage="inputoutput" /> <data name="KLORT" type="char" length="25"</pre> usage="inputoutput" /> <data name="KLTELK" type="char" length="3"</pre> usage="inputoutput" /> <data name="KLTEL" type="char" length="20"</pre> usage="inputoutput" /> <data name="KLFAXK" type="char" length="3"</pre>

Kapitel

5.2.5

Seite 116

Web-Programmierung Dialoge erstellen



Seite 117

usage="inputoutput" /> <data name="KLFAX" type="char" length="20"</pre> usage="inputoutput" /> <data name="KLFTFK" type="char" length="3"</pre> usage="inputoutput" /> <data name="KLFTF" type="char" length="20"</pre> usage="inputoutput" /> <data name="KLINTER" type="char" length="60"</pre> usage="inputoutput" /> <data name="KLEML" type="char" length="60"</pre> usage="inputoutput" /> <data name="KLULFNR" type="char" length="20"</pre> usage="inputoutput" /> <data name="KLBANK" type="char" length="30"</pre> usage="inputoutput" /> <data name="KLBLZ" type="char" length="8"</pre> usage="inputoutput" /> <data name="KLKTO" type="char" length="15"</pre> usage="inputoutput" /> <data name="KLDATN" type="char" length="10"</pre> usage="inputoutput" /> <data name="KLUSRN" type="char" length="10"</pre> usage="inputoutput" /> <data name="KLDATC" type="char" length="10"</pre> usage="inputoutput" /> <data name="KLUSRC" type="char" length="10"</pre> usage="inputoutput" /> </program> </pcml>

Ohne in die Details gehen zu müssen, erkennen Sie, dass es sich hier um eine Schnittstellenbeschreibung zu unserem Programm GETCSTSQL handelt.

Um diese Schnittstellenbeschreibung nutzen zu können, müssen wir das Script svklPreDetail erneut modifizieren.

Wir verzichten darauf, die Parameterdeklaration mittels i5\_programm\_prepare zu realisieren, und verwenden stattdessen die Funktion i5\_program\_ prepare\_PCML().



**Dialoge erstellen** 

5.2.5 Seite 118 Syntax:

```
resource i5_program_prepare_PCML (
array description
[, resource connection])
```

Die Funktion i5\_program\_prepare\_PCML erwartet zwei Parameter:

Description: Ein Array, das den Pfad zur PCML-Datei beschreibt. Dafür muss die PHP-Funktion file\_get\_contents() eingesetzt werden

Connection: Das Verbindungshandle, das die Funktion i5\_connect() erzeugt hat.

#### 5.2.5.18.1.4 Das Script svklDisplayPC.php

Dieses Script enthält die Dialoganzeige.

```
<?php

require("classes/InputElementField.php");

require("classes/InputElementHiddenField.php");

require("classes/InputElementRadiobutton.php");

require("classes/InputElementButton.php");
</pre>
```

Dieses Script enthält nur eine Besonderheit, die darin besteht, dass ein Combobox-Objekt eingebunden wurde.

Die Anzeige dieses Objekts wird analog zu den bereits bekannten Elementobjekten vorgenommen.



Kapitel

<pre>if (\$_GET['mode'] == "UPDATE") {</pre>	5.2.5
<pre>\$mode = "";</pre>	Seite 119
if (\$_GET['type'] == "KD"){	
<pre>\$header = "Kunde ändern";</pre>	
<pre>}elseif (\$_GET['type'] == "LF"){</pre>	
<pre>\$header = "Lieferant ändern";</pre>	
}	
<pre>}elseif (\$_GET['mode'] == "DELETE") {</pre>	
<pre>\$mode = "readonly";</pre>	
if (\$_GET['type'] == "KD"){	
<pre>\$header = "Kunde löschen";</pre>	
<pre>}elseif (\$_GET['type'] == "LF"){</pre>	
<pre>\$header = "Lieferant löschen";</pre>	
}	
}	
<pre>//echo \$_GET['type'];</pre>	
<pre>print("<form action='\"svklPostDetail.php\"' id='\"workKL\"' method='\"post\"' name='\"workKL\"' onsubmit='\"javascript:return' validatekl(this)\"=""></form>");     print("");</pre>	
<pre>print("");</pre>	
<pre>print("".\$header."");</pre>	
<pre>print("&gt; ");</pre>	
<pre>print("");</pre>	
nrint("	
print( <u> ), nrint("Firma / Identnummer / Unsere Kundenummer<!--</td--><td></td></u>	
<pre>print("");</pre>	
<pre>\$klfinr = new InputElementField(</pre>	
"klfinr",	
2,	
2,	





# Web-Programmierung Dialoge erstellen

5.2.5		\$_
Seite 120	POST['klfinr'],	
		1,
		"readonly");
	<pre>\$klfinr-&gt;displayInputText(); print(" ");</pre>	
	<pre>\$klidnr = new InputElementField(</pre>	
		"klidnr", _
		7,
		∕, \$
	POST['klidnr'],	*_
	/	2,
		"readonly");
	<pre>\$klidnr-&gt;displayInputText(); print(" "); if (\$ CET['type'] "LE") {</pre>	
	\$klulfnr = new InputElementField(	
		"klulfnr",
		20,
		20,
		\$_
	POSI['KIUITHE'],	3
		\$mode);
		- /
	<pre>\$klulfnr-&gt;displayInputText(); }</pre>	
	<pre>print("");</pre>	
	<pre>print("");</pre>	
	<pre>print("");</pre>	
	<pre>print("Anrede / Name");</pre>	
	print( <ta> ); \$k]Anrede = new InputFlementCombobox(</ta>	



	Ð
	pit
	Ka

5.2.5

Seite 121

"klanrede", //idname	
	1,
//size	4
//tabindex	.,
	\$mode,
//mode	\$_
POST['klankz'], //value	
POST['k]anrede'] //name	\$_
rosi kianiede j, //name	"selected",
//selected	
//kev	"ANKZ",
	\$system,
//RDBNAME	<b>f</b>
//Benutzer	\$user,
	<pre>\$passwd);</pre>
//Kennwort	
<pre>\$klAnrede-&gt;displayCombobox();</pre>	
<pre>print(" ");</pre>	
<pre>\$klname = new InputElementField(</pre>	
	"kIname",
	100,
	100,
	\$_
POST['kIname'],	
	5,
	<pre>\$mode);</pre>
<pre>\$klname-&gt;displayInputText();</pre>	
<pre>print("");</pre>	
print("");	
print("");	
<pre>print("");</pre>	







Web-Programmierung Dialoge erstellen POST['klplzp'], Seite 123 8, \$mode); \$klplzp->displayInputText(); print(" "); \$klortp = new InputElementField( "klortp", 25, 25, \$\_ POST['klortp'], 9, \$mode); \$klortp->displayInputText(); print(""); print(""); print(""); print(""); print("Land / Postleitzahl / Ort"); print(""); print(""); \$klland = new InputElementCombobox( "klland", 11 idname 1, //size 10, //tabindex \$mode, //mode \$\_ POST['kllandkz'], //value

apite



```
Web-Programmierung
```

## 5.2.5 Seite 124

\$\_ POST['klland'], //name "selected", //selected "LAND", //key \$system, //RDBNAME \$user, //Benutzer \$passwd); //Kennwort \$klland->displayCombobox(); print(" "); \$klplz = new InputElementField( "klplz", 7, 7, \$\_ POST['klplz'], 11, \$mode); \$klplz->displayInputText(); print(" "); \$klort = new InputElementField( "klort", 25, 25, \$\_ POST['klort'], 12, \$mode);



Web-Programmierung Dialoge erstellen



<pre>\$klort-&gt;displayInputText();</pre>		5.2.5
		Seite 125
print("");		
print("");		
<pre>print("");</pre>		
<pre>print("Telefon / Kurzwahl</pre>	");	
<pre>print("");</pre>		
<pre>\$kltel = new InputElementField(</pre>		
	"kltel",	
	20,	
	20,	
	\$	
POST['kltel'],		
	13,	
	<pre>\$mode);</pre>	
<pre>\$kltel-&gt;displayInputText();</pre>		
<pre>print(" ");</pre>		
<pre>\$kltelk = new InputElementField(</pre>		
	"kltelk",	
	3,	
	3,	
	\$_	
POST['kltelk'],		
	14,	
	<pre>\$mode);</pre>	
<pre>\$kltelk-&gt;displayInputText():</pre>		
<pre>print("");</pre>		
<pre>print("");</pre>		
<pre>print("");</pre>		
<pre>print("Fax / Kurzwahl");</pre>		
<pre>print("");</pre>		





Web-Programmierung Dialoge erstellen

5.2.5	<pre>\$klfax = new InputElementField(</pre>	
Seite 126		"klfax",
		20,
		20,
		\$_
	POST['klfax'],	
		15,
		<pre>\$mode);</pre>
	<pre>\$klfax-&gt;displayInputText();</pre>	
	<pre>print(" ");</pre>	
	<pre>\$klfaxk = new InputElementField(</pre>	
		"klfaxk",
		3,
		3,
		\$_
	POST['klfaxk'],	
		16,
		<pre>\$mode);</pre>
	(k)fork dicployIppytToyt().	
	<pre>\$kildxk-&gt;uisplayinputlext(); print("</pre>	
	p(1)(( )),	
	princ(  ),	
	print(""):	
	<pre>print("Handy / Kurzwahl"):</pre>	
	<pre>print("");</pre>	
	<pre>\$klftf = new InputElementField(</pre>	
		"klftf",
		20,
		20,
		\$_
	POST['klftf'],	
		17,
		<pre>\$mode);</pre>



len

5	Kapitel
5.2.	5

Seite 127

	<pre>\$klftf-&gt;displayInputText();</pre>	
	<pre>print(" ");</pre>	
	skrittk = new inputerementriera(	"klftfk"
		3.
		3,
		\$_
POST['k]	ftfk'],	
		18,
		<pre>\$mode);</pre>
	(k)ftfk_>dicployIpputToyt().	
	<pre>print(""):</pre>	
pr	<pre>int("");</pre>	
pr	<pre>int("");</pre>	
	<pre>print("Internet / E-Mail(allgemei)</pre>	n)");
	print("");	
	<pre>\$klinter = new InputElementField(</pre>	
		"klinter",
		60,
		60,
		\$_
POSIL'KI	inter'],	10
		19, \$mode):
		\$mode),
	<pre>\$klinter-&gt;displayInputText();</pre>	
	<pre>print(" ");</pre>	
	<pre>\$kleml = new InputElementField(</pre>	
		"kleml",
		60,
		60,
		\$_

Ergänzung 10/2009





Dialoge erstellen

5.2.5 Seite 128

POST['kleml'],

20, \$mode); \$kleml->displayInputText(); print(""); print(""); print(""); print("Bank / Bankleitzahl / Konto"); print(""); \$klbank = new InputElementField( "klbank", 30, 30, \$\_ POST['klbank'], 21, \$mode); \$klbank->displayInputText(); print(" "); \$klblz = new InputElementField( "klblz", 8, 8, \$\_ POST['klblz'], 22, \$mode); \$klblz->displayInputText(); print(" "); \$klkto = new InputElementField( "klkto", 15,



	Web-Program	nmierung 📕 🛓
	Dialog	ge erstellen
	15,	5.2.5
	\$_	Seite 129
POST['klkto'],	2.2	
	23, \$mode):	
	\$mode),	
<pre>\$klkto-&gt;displayInputText();</pre>		
print("");		
<pre>print("");</pre>		
<pre>print("");</pre>		
print("Datum Anlage / Benutzer	");	
<pre>print("");</pre>		
<pre>\$kldatn = new InputElementField(</pre>		
	"kldatn",	
	10,	
	10,	
	\$_	
POST['kldatn'],		
	24, "	
	readonly);	
<pre>\$kldatn-&gt;displayInputText();</pre>		
<pre>print(" ");</pre>		
<pre>\$klusrn = new InputElementField(</pre>		
	"klusrn",	
	10,	
	\$	
POST['k]usrn'],	*_	
	25,	
	"readonly");	

0

<b>D</b>	Web-Programmierung
Ka	Dialoge erstellen

```
5.2.5
                           $klusrn->displayInputText();
Seite 130
                           print("");
                         print("");
                         print("");
                           print("Datum Änderung / Benutzer");
                           print("");
                           $kldatc = new InputElementField(
                                                               "kldatc",
                                                                  10,
                                                                  10,
                                                                  $_
                   POST['kldatc'],
                                                                  26,
                                                                  "readonly");
                           $kldatc->displayInputText();
                           print(" ");
                           $klusrc = new InputElementField(
                                                               "klusrc",
                                                                 10,
                                                                  10,
                                                                  $_
                   POST['klusrc'],
                                                                  27,
                                                                  "readonly");
                           $klusrc->displayInputText();
                           $processingmode = new InputElementHiddenField(
                                                               "prcmode",
                                                                  $_
```

Dialoge erstellen







Ergänzung 10/2009



#### 5.2.5.18.1.5 Das Script svklPostDetail.php

?>

Das Script svklPostDetail.php führt dann die Nachbearbeitung aus, die darin besteht, die geänderten Daten wegzuschreiben bzw. die Löschanforderung durchzuführen.

<html> <head> <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" /> <LINK href="Styles/format.css" rel="stylesheet" type="text/css"> <script src="js/formhandling.js"</pre> type="text/javascript"></script> <title>Kunden und Lieferanten Nachricht</title> </head> <?php print("<body>"); require("includes/localSettings.php"); \$connectionstring = i5\_connect(\$host, \$user, \$passwd); if (!\$connectionstring) { die("Fehler in Verbindungsaufnahme (connect)".



i5_errormsg());	5.2.5
}	Seite 133
if (empty(\$_POST['klulfnr'])){	
<pre>\$_POST['klulfnr'] = " ";</pre>	
}	
\$inParms =array(	
"klfinr" => 1,	
"klidnr" => \$_POST['klidnr'],	
"klankz" => \$_POST['klankz'],	
"klanrede"=> " ",	
"klankz" => \$_POST['klankz'],	
"klstr" => \$_POST['klstr'],	
"klpstf" => \$_POST['klpstf'],	
<pre>"klplzp" =&gt; \$_POST['klplzp'],</pre>	
"klortp" => \$_POST['klortp'],	
"kllandkz"=> \$_POST['kllandkz'],	
"klland" => " ",	
"klplz" => \$_POST['klplz'],	
"klort" => \$_POST['klort'],	
"kltelk" => \$_POST['kltelk'],	
<pre>"kltel" =&gt; \$_POST['kltel'],</pre>	
"klfaxk" => \$_POST['klfaxk'],	
"klfax" => \$_POST['klfax'],	
"klftfk" => \$_POST['klftfk'],	
<pre>"klftf" =&gt; \$_POST['klftf'],</pre>	
"klinter" => \$_POST['klinter'],	
<pre>"kleml" =&gt; \$_POST['kleml'],</pre>	
"klulfnr" => \$_POST['klulfnr'],	
"klbank" => \$_POST['klbank'],	
"klblz" => \$_POST['klblz'],	
"klkto" => \$_POST['klkto'],	
"kldatn" => " ",	
"klusrn" => " ",	
"kldatc" => " ",	
"klusrc" => " "	
);	



Jetzt wird das Programm UPDCSTSQL vorbereitet und aufgerufen.

} elseif (\$\_POST['prcmode'] == "DELETE" ){

Handelt es sich um eine Löschanforderung, so wird dieser Zweig abgearbeitet.



Dialoge erstellen

Seite 135

```
$proghandle = i5_program_preparePCML(file_get_contents("pcml/
dltcustsql.pcml"));
if ($prog === FALSE){
    die("Programmvorbereitung gescheitert
        (prepare - DLT)".i5_errormsg());
}
frc = i5_program_call($proghandle, $inParms);
if ($rc === FALSE){
    die("Fehler im Programmaufruf (call -
        DLT)". i5_errormsg());
}
```

Auch hier wieder die Vorbereitung und die Ausführung des Programms.

```
$rc = i5_close($connectionstring);
if ($rc === FALSE){
    die("Fehler beim Beendigen der Verbindung (close)".
        i5_errormsg());
}
print ("Bearbeitung erfolgreich abgeschlossen.");
```

Sollte die Verarbeitung erfolgreich abgeschlossen worden sein, so wird eine entsprechende Meldung ausgegeben.

print("</body>");





**Web-Programmierung** Dialoge erstellen

5.2.5 Seite 136





Seite 13

## 5.2.5.19 **PHP und Opensource**

Nachdem wir uns bisher mit der PHP-Programmierung beschäftigt haben, möchte ich Ihnen in diesem Kapitel einen anderen Ansatz der Nutzung von PHP vorstellen.

Im Internet gibt es eine Vielzahl von fertigen PHP-Scripten, die wir als System i Anwender auf unserem System nutzen könnten.

Da gibt es viele Kleinfunktionen, die wir in unsere Programme einbauen könnten - Kalenderpopups, Texteditoren, Menügenerierungsprogramme etc. pp.

In einem ersten Schritt werden wir unsere bisher erstellten Scripte mit Hilfe eines Menügenerators, der frei verfügbar ist, zu einer Anwendung verbinden.

Wenden wir uns jedoch zunächst dem Menügenerator zu.

## 5.2.5.19.1 Menüs mit CoolJs generieren

Bis zu diesem Zeitpunkt haben wir eine Folge von einzelnen Scripten erzeugt, von denen einige logisch aufeinander folgen. Wenn wir uns jedoch vorstellen, wie eine große Anwendung sich darstellt, so werden wir nach kurzer Zeit bemerken, dass viele Funktionen existieren, die quasi auf der gleichen funktionalen Ebene genutzt werden können.

Kurz: Wir brauchen ein Menü, mit dem wir in die jeweiligen Anwendungszweige gelangen.

Wir könnten uns jetzt sicherlich hinsetzen und mit PHP und/oder JS ein Menü selbst erstellen, es scheint mir jedoch einfacher einen Menügenerator für derartige Tätigkeiten zu verwenden.

Ich habe auf der Seite http://javascript.cooldev.com/scripts/cooltree/ einen derartigen Generator gefunden, den ich verwendet habe, um für unsere Anwendung ein Menü zu erstellen.

Der CoolJs Menügenerator wird in zwei Varianten zur Verfügung gestellt.

- Kostenfrei: funktionell reduziert, jedoch in unserem Zusammenhang gut zu verwenden
- Kostenpflichtig: funktionell komplett ausgebaut

Hier sehen Sie eine Hardcopy des CoolJs Menügenerators.







Dialoge erstellen





#### Coolbuilder

Mit diesem Tool erzeugen Sie Menüstrukturen, die sich als Verzeichnisbaum abbilden.

Der Coolbuilder erzeugt einige Objekte, die Sie in Ihre Seiten einarbeiten können. Ich habe zu diesem Zweck eine Seite menu.html erzeugt.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type"

content="text/html; charset=iso-8859-1" />

<title>Unbenanntes Dokument</title>

<link href="format.css" rel="stylesheet"

type="text/css" />

<link href="tree_styles.css" rel="stylesheet"

type="text/css">
```



Web-Programmierung	
Dialoge erstellen	Ϋ́
CSS mit dem Namen tree_styles.css, mit dessen	5.2.5

Seite 139

Der Coolbuilder erzeugt ein CSS mit dem Namen tree\_styles.css, mit dessen Hilfe der Verzeichnisbaum formatiert wird.

```
<script type="text/javascript"
src="cooltree.js"></script>
<script type="text/javascript"
src="tree_nodes.js"></script>
<script type="text/javascript"
src="tree_format.js"></script>
```

Weiterhin erzeugt der Coolbuilder drei JS-Dateien, die ebenfalls in das Script aufzunehmen sind:

- cooltree.js
- tree\_nodes.js
- tree\_format.js

```
</head>
<body bgcolor="#CCCCCC">
<!-- <p class="Tableheader">Navigation -->
<script type="text/javascript">
var tree1 =
new COOLjsTree("tree1", TREE_NODES,
TREE_FORMAT);
</script>
```

Des Weiteren benötigen Sie einen Scriptaufruf der Funktion COOLjsTree.

</body>
</html>





Dialoge erstellen

5.2.5 Seite 140 Diese Startseite habe ich dann in einen HTML-Frame eingebettet, dem ich dem Namen index.html gegeben habe, denn unser Script signon.php ruft nach erfolgreicher Anmeldung eine Seite mit dem Namen index.html auf.

Hier die Seite index.html:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1" />
<title>EPOS Schulungsverwaltung</title>
</head>
<frameset rows="80,*" cols="*"</pre>
          frameborder="no" border="0"
         framespacing="0">
      <frame <pre>src="header.html"
          name="topFrame" scrolling="No"
          noresize="noresize" id="topFrame"
         title="topFrame" />
      <frameset cols="200,*" frameborder="no"
          border="1" framespacing="0">
      <frame src="menu.html" name="leftFrame"</pre>
          scrolling="No" resize="noresize"
          id="leftFrame" title="leftFrame" />
      <frame src="stats.php" name="mainFrame"</pre>
          id="mainFrame" title="mainFrame" />
      </frameset>
      </frameset>
<noframes>
<body>
</body>
</noframes>
</html>
```

Web-Programmierung	

Seite 14

In diesem Frame sind zwei HTML- und ein PHP-Dokument eingebunden (Markierung!).

Header.html enthält nur die Überschrift Stammdatenverwaltung.

Menu.html kennen Sie bereits. Es enthält das mit Coolbuilder generierte Menü.

Stats.php kennen Sie auch. Es enthält die Anmeldedaten und Systemwerte.

Der gesamte Dialog sieht jetzt wie folgt aus:

#### 1. Anmeldung



## Am System i anmelden

#### 2. Menüanzeige

EPOS Schulungsverwaltung		
		Stammdatenverwaltung
Stammdaten Stammdaten Kunden und Lieferanten Kunden und Lieferanten Storgänge Steatellungen Historie	Datum: 091107 Zeit: 123000799 Benutzer: LUTTKUS	

Coolbuilder Menü wird angezeigt

## 3. Auswahl Kunden und Lieferanten

EPOS Schulungsverwaltung									
				Stammdate	nverwa	itu	ng		
Stammdaten	Kunden un	d Liefera	nten sucher			_			
Ansprechpartner	Suchbegriff			ds					
vorgange Bestellungen	Kunden / Lie	eferanten		• C					
3) Historie		Suchen Initialisieren							
	Funktioner	Identnr.	Name		Land	PLZ	Ort	Strasse	Telefon Kunde
	🔍 🔕 词	1014500	Kraft Foods I	Deutschland GmbH & Co. KG	Deutschland	28199	Bremen	Langemarckstr. 4-20	03.12.02
		1027400	Harrods Knig	ghtsbridge Limited	Groß Britannien	SWIX 7X	London	Knightsbridge	29.05.02
	s 😒 🕞	1042000	Berendsohn	AG	Deutschland	22559	Hamburg	Alte Sülldorfer Landstr. 400	29.05.02
	🖸 🔕 💽	1064000	JD Edwards	Deutschland GmbH	Deutschland	64546	Mörfelden/Frankfurf	Waldeckerstrasse 11	11.12.02
	S 8 5	1089900	Wort+Bild V	erlag Konradshöhe GmbH & Co.	Deutschland	82065	Baierbrunn	Konradshöhe	12.12.02
	S 😣 🗔	1142600	ADS Inform	atik Hard- u. Software-Beratung	Deutschland	66693	Mettlach	Goetheweg 8	12.12.02
	<u>s</u> <u>s</u> <u>s</u>	1202600	Smith + Nep Medizintechr	nhew Richards GmbH nik	Deutschland	78532	Tuttlingen	Alemannenstr. 14	11.02.03
	🖾 🔕 词	1205800	DIFA Deutso	he Immobilien Fonds AG	Deutschland	20354	Hamburg	Valentinskamp 20	29.05.02
	🖾 🔕 词	1224500	HUDSON In	ern. Vertr. GmbH	Deutschland	70565	Stuttgart	Am Wallgraben 45	29.05.02
		1001000	La lank Kalling	andeteinunglig Carlel I.O. Ca	Day day he have a	24560	Kalkan Unakan	Deservate alters Chr. 1.4	20.05.02

Anzeige der Daten

Ergänzung 11/2009





Kunden ändern

Damit soll die Programmierung zunächst einmal abgeschlossen sein. Wenden wir uns jetzt der Nutzung von Open Source Anwendungen zu.


# Dialoge erstellen

apitel

5.2.5.19.2 Die Installation von MySQL

Die Voraussetzungen für die Installation vom MySQL im i5/OS entsprechen denen der PHP Installation.

Folgende Lizenzprogramme müssen installiert sein:

V5R3:

Portable App Solutions Environment	33	5722ss1
Qshell	30	5722ss1
System Openness Includes	13	5722ss1
Digital Certificate Manager	34	5722ss1
Cryptographic Access Provider	*base	5722ac3
IBM Portable Utilities For I5/OS	*base	5733sc1
OpenSSH ,OpenSST,Zlib	1	5733sc1
http Server	*base	5722dg1

V5R4:

Portable App Solutions Environment	33	5722ss1
Qshell	30	5722ss1
System Openness Includes	13	5722ss1
Digital Certificate Manager	34	5722ss1
CCA Cryptographic Service Provider	35	5722ss1
IBM Portable Utilities For I5/OS	*base	5733sc1
OpenSSH ,OpenSST,Zlib	1	5733sc1
http Server	*base	5722dg1



# 5.2.5 Seite 144

#### 5.2.5.19.2.1 Download MySQL



Laden Sie zuerst MySQL von der MySQL-Homepage herunter.

http://dev.mysql.com/downloads/mysql/5.1.html#aix

IBM AIX downloads (platform notes)		
AIX 5.2 (POWER, 64-bit)	5.1.21-beta 59.3M	Download   Pick a mirror
	MD5: 6c52a9bda3c	c84864fef096e00ddd42a   Signature
AIX 5.2 (POWER, 32-bit)	5.1.21-beta 58.9M	Download   Pick a mirror
	MD5: 9b68f0640c7	315bca6fa70d8ad682efa   Signature

Download

Achten Sie darauf, dass Sie die Version für AIX – 64 bit herunterladen.

# 5.2.5.19.2.2 Benutzerprofil erstellen

Erstellen Sie dann im i5/OS das Benutzerprofil MYSQL:

Benutzerprofil ändern (	CHGUSRPRF)
Auswahl eingeben und Eingabetaste drücken.	
Benutzerprofil > MYSQL	Name
Benutzerkennwort <u>*SAME</u>	Zeichenwert, *SAME, *NONE
Kennwort auf abgelaufen setzen <u>*NO</u>	*SAME, *NO, *YES
Status <u>*ENABLED</u>	*SAME, *ENABLED, *DISABLED
Benutzerklasse <u>*SECOFR</u>	*SAME, *USER, *SYSOPR
Unterstützungsstufe <u>*SYSVAL</u>	*SAME, *SYSVAL, *BASIC
Aktuelle Bibliothek <u>*CRTDFT</u>	Name, *SAME, *CRTDFT
Aufzurufendes Startprogramm <u>*NONE</u>	Name, *SAME, *NONE
Bibliothek	Name, *LIBL, *CURLIB
Anfangsmenü MAIN	Name, *SAME, *SIGNOFF
Bibliothek	Name, *LIBL, *CURLIB
Möglichkeiten einschränken <u>*NO</u>	*SAME, *NO, *PARTIAL, *YES
Text 'Beschreibung' <u>'Zend - MyS</u>	ql user profile'

CRTUSRPRF

Password: \*NONE

UsrCls: \*SECOFR





Seite 145

# 5.2.5.19.1.3 Installationsdateien entpacken

Nach dem Download der Installationsdateien erhalten Sie je nach Version eine gepackte Datei:

mysql-5.1.20-beta-aix5.2-powerpc-64bit.tar.gz

Diese können Sie mit einem Entpackungstool wie Winzip entpacken.

# 5.2.5.19.2.4 Entpackte Dateien auf das System i kopieren

4. Nach dem Entpacken erhalten Sie z. B. eine Datei folgenden Namens:

mysql-5.1.20-beta-aix5.2-powerpc-64bit.tar

Mittels einer i5/OS Netserver-Freigabe kopiere ich diese Datei in einen IFS Pfad, wie z. B. /usr/local.

Dieser Pfad ist nicht zwingend notwendig, Sie können auch andere Pfade verwenden.

# 5.2.5.19.2.5 PASE-Terminal öffnen

In einer 5250-Sitzung rufe ich das PASE-Terminal mit folgendem Befehl auf:



CALL QP2TERM mit QSECOFR

Mit folgendem Befehl navigiere ich dann in den Pfad /usr/local.

Ergänzung 11/2009



# 5.2.5 Seite 146

#### 5.2.5.19.2.6 Tar-File entpacken

Das in das IFS übertragene Tar-File entpacke ich dann in der PASE-Shell mit folgendem Befehl:

tar -xvf mysql-5.1.20-beta-aix5.2-powerpc-64bit.tar

Es werden dann Meldungen folgender Art angezeigt.

Meldungen:

```
x mysql-5.1.20-beta-aix5.2-powerpc-64bit/COPYING, 19071
Bytes, 38 Datenträgerblöcke.
x mysql-5.1.20-beta-aix5.2-powerpc-64bit/README, 1410 Bytes,
3 Datenträgerblöcke.
x mysql-5.1.20-beta-aix5.2-powerpc-64bit/INSTALL-BINARY, 8523
Bytes, 17 Datenträgerblöcke.
x mysql-5.1.20-beta-aix5.2-powerpc-64bit/EXCEPTIONS-CLIENT,
5139 Bytes, 11 Datenträgerblöcke.
x mysql-5.1.20-beta-aix5.2-powerpc-64bit/configure, 801
Bytes, 2 Datenträgerblöcke.
#
```

# 5.2.5.19.2.7 Logischen Link erstellen

Beim Entpacken wird ein Verzeichnis mit z.B. folgendem Namen erzeugt: mysql-5.1.20-beta-aix5.2-powerpc-64bit

Damit ich in der Folge mit einem einfacheren Namen arbeiten kann erzeuge ich einen logischen Link mit folgendem Befehl:

ln -s mysql-5.1.20-beta-aix5.2-powerpc-64bit mysql5\_1\_20

Ich wechsle dann in den Link mysql5\_1\_20 mit dem Befehl:

cd /mysql5\_1\_20





### 5.2.5.19.2.8 Installationsscript ausführen

Jetzt muss das MySQL Installationsscript ausgeführt werden:

scripts/mysql\_install\_db --user=mysql

```
Meldungen:
scripts/mysql_install_db --user=mysql
Installing MySQL system tables...
070807 19:34:11 [Warning] Setting lower_case_table_names=2
because file system for /usr/local/mysql-5.1.20-beta-aix5.2-
powerpc-64
bit/data/ is case insensitive
setgroups: Invalid argument
070807 19:34:11 [Note] Plugin 'InnoDB' disabled by command
line option
070807 19:34:11 [Warning] ./bin/mysqld: unknown option
,--loose-skip-ndbcluster'
OK
Filling help tables...
070807 19:34:13 [Warning] Setting lower_case_table_names=2
because file system for /usr/local/mysql-5.1.20-beta-aix5.2-
powerpc-64
bit/data/ is case insensitive
setgroups: Invalid argument
070807 19:34:13 [Note] Plugin 'InnoDB' disabled by command
line option
070807 19:34:13 [Warning] ./bin/mysqld: unknown option
'--loose-skip-ndbcluster'
```

Ergänzung 11/2009





# Web-Programmierung Dialoge erstellen

5.2.5 Seite 148	To start mysqld at boot time you have to copy support-files/mysql.server to the right place for your system PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER ! To do so, start the server, then issue the following com- mands: ./bin/mysqladmin -u root password ,new-password' ./bin/mysqladmin -u root -h EPOSI501.EPOSHBINT.LOCAL password 'new-password' See the manual for more instructions. You can start the MySQL daemon with: cd . ; ./bin/mysqld_safe &
	You can test the MySQL daemon with mysql-test-run.pl cd mysql-test ; perl mysql-test-run.pl
	Please report any problems with the ./bin/mysqlbug script!
	The latest information about MySQL is available on the web at http://www.mysql.com
	Support MySQL by buying support/licenses at http://shop. mysql.com
	#





Saita 140

# 5.2.5.19.2.9 MySQL Probestarten

Mit folgendem Befehl führen Sie einen Probestart des MySQL-Servers durch: bin/mysqld\_safe --user=mysql

Meldungen:

Verwendung: touch [-amcf] [mmtthhmm[jj]] [-t [[CC]JJ] MMTThhmm[.SS]] [-r Bezugsdatei] Datei ... Verwendung: chown -fhR {Benutzername|UID}[:{Gruppenname|GID}] Datei ... chown -R [-H | -L | -P] [-f] {Benutzername|UID} [:{Gruppenname|GID}] Datei ... 070807 19:38:55 mysqld\_safe Starting mysqld daemon with databases from /usr/local/mysql5\_1\_20/data

Achtung, bei diesem Start wird MySQL in der Bildschirmsitzung ausgeführt! Für eine produktive Umgebung ist das sicherlich nicht wünschenswert. Aber es zeigt Ihnen die Funktionsfähigkeit der Installation.

# 5.2.5.19.3 MySQL produktiv nutzen

Ergänzung 11/2009

Neben Programmiererhilstools existieren jedoch auch fertige "große" PHP-Anwendungen, wie Content Management Systeme (CMS), HelpDesk Systeme, Wikis, Blogs und Customer Relationship System (CRS), die nur darauf warten von uns heruntergeladen, installiert und angewendet zu werden.

Da viele dieser Programme nicht mit der DB2-Datenbank arbeiten, sondern mit MySQL, eine Umstellung der Anwendungen auf DB2 möglich, jedoch auch arbeitsreich wäre, sollten wir von der Möglichkeit Gebrauch machen, MySQL auf dem System i zu installieren.

Deshalb werde ich Ihnen kurz erklären, wie MySQL auf dem System i installiert wird.

Danach werde ich Ihnen beschreiben, wie zwei populäre PHP-Anwendungen auf dem System i installiert werden:

- Mediawiki, ein Wiki, das sich gut für die hausinterne Dokumentation einsetzen lässt.
- SugarCRM, ein PHP-basiertes Customer Relashionship System.



Dialoge erstellen

5.2.5 Seite 150 Damit MySQL auch produktiv betrieben werden kann, habe ich für den MySQL-Server eine eigene Jobumgebung eingerichtet.

Diese besteht aus:

- Jobwarteschlange: \*JOBQ
- Subsystem: \* SBSD
- Ausgabewarteschlange: \*OUTQ
- Jobbeschreibung:\*JOBD
- Jobklasse: \*CLS

All diese Objekte habe ich in einer eigenen Bibliothek MySQL gespeichert.

Im Folgenden zeige ich Ihnen die Konfigurationsdialoge der genannten Objekte.

### 5.2.5.19.3.1 Jobwarteschlange MySQL

	Jobwarteschlange erstellen	(CRTJOBQ)
Auswahl eingeben und	Eingabetaste drücken.	
Jobwarteschlange	JOBQ >	MYSQL
Bibliothek	· · · · · · · · · · · · · · · · · · ·	MYSQL
Text 'Beschreibung'	TEXT	*BLANK



#### 5.2.5.19.3.2 Ausgabewarteschlange MySQL

	Ausgabewarteschlange ändern	(CHGOUTQ)
Auswahl eingeben und	Eingabetaste drücken.	
Ausgabewarteschlange Bibliothek Max. Größe der Spool-		Name Name, *LIBL, *CURLIB
Anzahl Seiten	<u>*NONE</u>	Zahl, *SAME, *NONE
Startzeit	<u> </u>	Zeit
Endzeit		Zeit
+ für weitere We Dateireihenfolge in W	erte _ Warteschl. <u>*FIFO</u>	*SAME, *JOBNBR, *FIFO

Attribute \*OUTQ





Jobbeschreibung ändern (CHGJOBD) Auswahl eingeben und Eingabetaste drücken.
Auswahl eingeben und Eingabetaste drücken.
Jobbeschreibung JOBD > MYSQL
Bibliothek > MYSQL
Jobwarteschlange JOBQ 🚬 🛛 MYSQL
Bibliothek
Jobpriorität (in JOBQ) JOBPTY <u>5</u>
Ausgabepriorität (in OUTQ) OUTPTY <u>5</u>
Druckeinheit PRTDEV <u>*USRPRF</u>
Ausgabewarteschlange OUTQ <u>*USRPRF</u>
Bibliothek
Text 'Beschreibung' TEXT <u>*BLANK</u>

Attribute \*JOBD 1

5.2.5.19.2.3

	Zusätzliche	Parameter
Benutzer		USER 🔨 🗡 MYSQL
Berechnungscode		ACGCDE <u>*USRPRF</u>

Attribute \*JOBD 2

### 5.2.5.19.3.4 Jobklasse MySQL

		k	<b>(</b> ];	as	se ändern (CHGC	LS)
Auswahl eingeben und Eing	gab	e	tas	st	e drücken.	
Klasse	•			•	CLS >	MYSQL
Bibliothek	i.					MYSQL
Ausführungspriorität	•				RUNPTY	50
Zeitscheibe		8			TIMESLICE	10000
Auswählbar für Auslagern	•	•	a.	5	PURGE 🥄 🖊	*YES
Standardwartezeit	•				DFTWAIT 🗸	30
Maximale CPU-Zeit			8	×.	CPUTIME	*NOMAX
Max. temporärer Speicher	•				MAXTMPSTG	*NOMAX
Maximale Anzahl Threads	-				MAXTHD	*NOMAX
Text 'Beschreibung'					TEXT	*BLANK

Attribute \*CLS



apitel

**Dialoge erstellen** 

# 5.2.5 Seite 152 Subsystem MySQL

			10.0743			101101								23	
Folgenummer des	Lei	twe	ege	≥ır	nti	rag	js	•		•		•	1		9999
Programm	6 - 16	×	5	•	•		9	c)	i)	×.	s <b>.</b>	e)	i,	:	QCMD
Bibliothek .	9 B	۰	2		s		2	10	8		e.	•		1	QSYS
Klasse	14	÷	а,	i.	÷	4	5	2		s.	5	2		3	MYSQL
Bibliothek .	0.0		9	e						a.		•9		1	MYSQL
Maximale Anzahl	akt	ive	≥r	Le	ei '	twe	ege	sch	nri	t	te	•	÷	1	*NOMAX
Pool-ID	e 8		5	e			4	e			5	e		:	1
Vergleichswert			a					•		æ				:	*ANY 🥌

Attribute \*SBSD: Leitwegeintrag

# 5.2.5.19.3.6 Start des MySQL-Servers

Um den MySQL-Server zu starten, habe ich dann folgendes CL-Programm geschrieben und aufgerufen:

strmysql:	pgm	-
	STRSBS	SBSD (MYSQL/MYSQL)
	MONMSG	CPF0000
	SBMJOB	CMD (CALL PGM (QP2SHELL)
	PARM('∕u≘	sr/local/mysql/bin/mysqld_safe' +
		'user=mysql'))
		JOB (MYSQLD)
		JOBD (MYSQL/MYSQL)
		LANGID (ENU) CNTRYID (US)
		OUTQ (MYSQL/MYSQL) USER (MYSQL) RTGDTA (MYSQL)
exit:		
	endpgm	

MySQL Startprogramm

Nach dem Aufruf des Startprogramms prüfe ich mit WRKACTJOB den hoffentlich erfolgreichen Start.

Auswahl eingeben und Eingabetaste drücken.							
2=	Ändern 3=Anha	lten	4=Beer	nden	5=Arbe	iten mit	6=Freigeben
7=Nachricht anzeigen 8=Mit Spool-Dateien arb					n arbeiten	13=Unterbrechen	
		Aktuel	ler				
Opt	Subsystem/Job	Benutz	er	Art	CPU %	Funktion	Status
	MYSQL	QSYS		SBS	0,0		DEQW
	MYSQLD	MYSQL		BCH	0,0	PGM-sh	THDW
	MYSQLD	MYSQL		BCI	0,0	PGM-mysqld	SELW

MySQL Serverjobs



# 5.2.5.19.3.7 **Beendigung des MySQL-Servers**

Zum Beenden des SQL-Servers habe ich folgendes CL-Programm geschrieben:

```
endmysql:
             pgm
             SBMJOB CMD (CALL PGM (QP2SHELL)
                               PARM('/usr/local/mysql/bin/mysqladmin'
                                    '--user=root' 'shutdown'))
                     JOB (ENDMYSQL) JOBD (MYSQL/MYSQL) LANGID (ENU) CNTRYID
             DLYJOB
                         DLY (30)
                         SBS (MYSQL) OPTION (*IMMED)
             ENDSBS
             MONMSG
                         CPF0000
exit:
             endpgm
```

*MySQL* beenden

### 5.2.5.19.3.8 Administration

Um mit fernen Administrationsclients den MySQL-Server administrieren zu können, müssen Sie diese Clients entsprechend berechtigen.

Öffnen Sie dazu das PASE Terminal.

Geben Sie im MySQL-Server Verzeichnis /bin den Befehl mysql ein.

Am mysql> Prompt geben Sie dann folgenden Befehl ein:

```
GRANT ALL PRIVILIGES ON *.*
TO ,NeuerDbBenutzer'@'%'
IDENTIFIED BY , password'
WITH GRANT OPTION;
```



Seite 153



5.2.5 Seite 154 Von der MySQL Homepage habe ich mir dann die grafischen Administrationstools für MySQL heruntergeladen und auf meinem Windows PC installiert.

Iministrator	
Connect to MySQL Server Instance	
Server Host: 172.16.110.22	Port: 3306
Username: Juttkus	2
Password: *****	

#### Anmeldung

In dem gezeigten Dialog gebe ich lediglich die IP-Adresse des System i und die Anmeldedaten des soeben erzeugten Benutzers ein.

Jetzt öffnet sich das grafische Admin-Tool.

MySQL Administrator - lut	tkus@172.16.110.22:3306	
File Edit View Tools Winde	ow Help	
Server Information Service Control Startup Variables	Server status: MySQL Server is running.	MySQL
Server Connections Health Server Logs Replication Status Backup Restore Catalogs	Connected to MySQL Server Instance Username: luttkus Hostname: 172.16.110.22 Port: 3306 Server Information MySQL Version: MySQL 5.0.21-standard via TCP/IP Network Name: eposi501.eposhbint.local IP: 172.16.110.22	
4 min 4	Client Information Version: MySQL Client Version 5.0.11 Network Name: GF-REMOTE-01 IP: 172.16.200.1 Operating System: Windows XP Hardware: Intel(R) Pentium(R) M processor 2.13GHz, 2.0 GB RAM	

#### Admin-Tool 1

Alternativ zu dem gezeigten Windows-Programm gibt es auch eine weit verbreitete PHP-Anwendung für die Administration von Mysql: PhpMyAdmin.

Laden Sie PhpMyAdmin aus dem Internet herunter. Um eine Downloadseite zu finden, suchen Sie einfach in Google nach den Begriffen PhpMyAdmin und download.





Sie erhalten dann eine gepackte Datei.

Diese entpacken Sie auf Ihrem PC und kopieren die entpackten Objekte in das Verzeichnis /www/zendcore/phpyadmin.

Rufen Sie sodann PhpMyAdmin unter folgender URL auf:

http://<PHP-Server-IP-Adresse/phpmyadmin



Admin-Tool 2

#### 5.2.5.19.4 Mediawiki installieren

Wenn Sie MySQL auf dem System i installiert haben, benötigen Sie jetzt nur eine Kurzanleitung, um Mediawiki zu installieren:

- 1. Von http://www.mediawiki.com MediaWiki (mediawiki-x.x.tar.gz) herunterladen.
- 2. mediawiki-x.x.x.tar.gz mit Zip-Werkzeug entpacken.
- 3. mediawiki-x.x.tar von PC zum System i (\www\zendcore\htdocs) kopieren.
- 4. System i: PASE-Shell aufrufen CALL QP2TERM
- 5. cd /www/zendcore/htdocs
- 6. mediawiki-x.x.tar entpacken: tar -xvf mediawiki-1.9.3.tar
- 7. cd mediawiki-x.x.x
- 8. Berechtigungen für Verzeichnis config ändern: chmod a+w config
- 8.1. Optional: Mediawiki-Root-Verzeichnis umbenennen: Vorschlag: Mediawiki







Dialoge erstellen

# 5.2.5 Seite 156

- 9. Browser öffnen: http:/<Hostname>:Port/mediawiki-x.x.x
- 9.1 Bei geändertem Root-Verzeichnis: http:/<Hostname>:Port/mediawiki
  - 10. Dem Dialog folgen: Mediawiki einen CI-Namen geben, Kennwörter, wo notwendig, erstellen.
  - 11. Fertig! Browser öffnen:

http:/<Hostname>:Port/mediawiki-1.9.3 bzw.

http:/<Hostname>:Port/mediawiki

12. Anpassungen an MediaWiki-Oberfläche vornehmen.



Dokumentationen anlegen



# 5.2.5.19.5 SugarCRM installieren

### 5.2.5.19.5.1 Voraussetzungen:

- 1. MySQL muss auf dem System i installiert und funktionsfähig sein.
- 2. Gegebenenfalls PHP aktualisieren:

### 5.2.5.19.5.2 Installation

- 1. Von http://www.sugarcrm.com die Open Source Version Sugar Open Source 4.5.1 (SugarOS-4.5.1e.zip) herunterladen.
- 1.1 Das deutsche Language Pack herunterladen (SugarCRM\_langpack\_ge\_ge\_4-5-1.zip).
- 2. Mit Zip-Programm SugarOS-4.5.1e.zip auf dem PC entpacken.
- 3. Die entpackten Dateien vom PC nach \www\zendcore\htdocs\sugarcrm auf dem System i kopieren.
- 4. PHP.ini modifizieren (/usr/local/Zend/Core/etc):

```
[mbstring]
mbstring.language = Japanese
mbstring.internal_encoding = UTF8
mbstring.http_input = auto
mbstring.http_output = pass
mbstring.encoding_translation = On
mbstring.detect_order = auto
mbstring.substitute_character = none;
mbstring.func_overload = 0
upload_max_filesize=6M
memory_limit=64M
session.save_path = "/tmp"
```

Beachten Sie, dass auf /tmp für alle Schreibrechte vergeben sind.

Dialoge erstellen



# **5.2.5** Seite 157





Dialoge erstellen

# 5.2.5 Seite 158

#### 5. Browser mit folgender URL aufrufen: http://<Hostname>:port/sugarcrm/



#### Installation 1

Mit Start in den nächsten Dialog.



Installation 2





Seite 159

# Mit Weiter zum nächsten Dialog

Schritt 2: System Check Akzeptanz	SUGARCRM.
Eine fehlerfreie SugarCRM Installation ist nur bei Bitte sorgen Sie ggf. fļr entsprechende Probler	i grļnen Punkten des Systemchecks gewÄ≋hrleistet. mlĶsung.
Komponente	Status
PHP-Version	Nicht unterstĽtzte PHP Version installiert: (ver 5.1.6)
MySQL Database	ок
MB Strings Module	ок
XML Parsing	OK
PHP Safe Mode steht auf Off	ок
PHP erlaubt Call Time Pass Reference Turned On	ок
PHP Memory-BeschrĤnkung >= 32M	OK (64M)
Beschreibbare SugarCRM-Konfigurationsdatei (config.php)	ок
Beschreibbares Kunden-Verzeichnis	ок
Beschreibbare Module, Sub-Verzeichnisse und Dateien	ок
Beschreibbare Daten Sub-Verzeichnisse	ок
Schreibbarer Cache - Sub-Verzeichnisse	ОК
Speicherpfad fļr beschreibbare Sessions (/tmp)	ок
Optional Components	Status
IMAP Module	Not found: InboundEmail and Campaigns (Email) require the IMAP libraries. Neither will be functional.
cURL Bibliothek	Nicht gefunden: Sugar Zeitplaner wird nur eingeschrĤnkt funktionieren.
ZLIB Compression Module	ОК
Hinweis: Thre php-Konfigu /usr/local/	rationsdatei (php.ini) fiinden Sie unter: Zend/core/etc/php.ini
Hil	fe Nochmals Ä4berprÄ4fen ZurÄ4ck Weiter

### Installation 3

Aktivieren Sie in der php.ini gegebenenfalls folgende Erweiterungen:

extension=curl.so extension=imap.so

Das ist nur notwendig, wenn Mailempfang und Terminplanung von Sugar-CRM eingesetzt werden sollen.





Dialoge erstellen

# 5.2.5 Seite 160

Wenn Status insgesamt grün ist, mit Weiter weitergehen.

Schritt 3: Datenbank Konfiguration		SUGARCRM.	
Bitte geben Sie Ihre Datenbank-Konfigurationsdaten ein. Fall S Standardeinstellungen zu nutzen.	ie diese nicht ke	nnen, empfehlen wir, die	
* Erforderliche Angaben			
Datenbank Konfiguration			
* Host-Name	localhost		
* Datenbank-Name	sugarcrm	🔽 Datenbank erstellen	
* Datenbank-Benutzername	sugarcrm	Benutzer erstellen	
Datenbank-Passwort	*****		
Datenbank-Passwort wiederholen	*****	- N	
Bestehende Sugar Datenbank-Tabellen wiederherstellen? Achtung: Alle Sugar Daten werden gelĶscht falls diese Checkbox aktiviert ist.		2,	
Datenbank mit Demo-Daten fļllen?			
Datenbank-Account ist ein Privilegiert Benutzer?	Г		
Priviligierter Datenbank-Benutzername Der priviligierte Datenbank-Benutzer muss das Recht haben, eine Datenbank zu erstellen, Tabellen zu lä¶schen etc. Der priviligierte Datenbank-Benutzer wird nur genutzt, um währhend der Installation auf die Datenbank zugreifen zu kä¶nnen.	root	3.	
Passwort fļr priviligierten Datenbank-Benutzer			
~		Hilfe Zurļck Weiter	

#### Installation 4

# Mit Weiter in den nächsten Dialog gehen.

Schritt 4: Seiten-Konfiguration	SUGARCRM.
Bitte geben Sie Informationen zur Konfiguration Ihrer Installation e empfehlen wir Ihnen, die die Standardeinstellungen zu nutzen.	ein. Falls Ihnen Informationen fehlen,
*Erforderliche Angaben	
Seiten-Konfiguration	
* URL der Sugar-Installation	http://172.16.110.22/sugarcrm
<sup>6</sup> System Name This will be displayed in the Title Bar of users who visit this SugarCRM installation	eposi501.eposgmbh.com
<sup>1</sup> Sugar Admin Passwort Achtung: Dies wird alle Admin-PasswĶrter frļherer Installation ļberschreiben.	
* Wiederholung Sugar Admin Passwort	жжжжя
Update-Konfiguration	
Send Anonymous Usage Statistics? If checked, Sugar will send anonymous statistics about your installation to SugarCRM Inc. every time your system checks for new versions. This information will help us better understand how the application is used and guide improvements to the product.	<b>J</b>
Sugar Updates aktivieren? Wenn der Punkt aktiviert ist, Ľbermittelt Ihr System anonyme Informationen zu Ihrer Installation an SugarCRM Inc. Dies hift uns, unsere Produkte gemĤğ Ihren Anforderungen weiterzuentwickeln. Im Gegenzug erhĤlt der Administrator Hinweise auf Updates oder neue Versionen.	<b>न</b>
Erweiterte Sicherheitseinstellungen	
Standard-Einstellungen nutzen?	
	Hilfe ZurA¼ick Weiter

Installation 5





5.2.5

Seite 161

# Mit Weiter in den nächsten Dialog gehen.

Schritt 5: Locale Settings	SUGARCRM.
Sie kĶnnen hier ein weiteres Sprachpaket installieren. And nÄ∝chsten Schritt.	ernfalls kommen Sie mit "Weiter" zum
Locale Settings	
Adjust your SugarCRM Locale settings below.	
User Interface	
Default Date Format:	23.12.2006 💽 🗲
Default Time Format:	23:00 💌
Default Language:	US English 🖃
Default Name Format:	sfl Dr. David Livingstone "s" Salutation
	"f" First Name "I" Last Name
Outbound Email Character Set	
Set this to the character set most commonly used in your locale:	ISO-8859-1 (Western European and US)
Export Settings	
Import/Export Character Set (Email, .csv, vCard, PDF, data import):	CP1252 (MS Western European & US)
Export (.csv) Delimiter:	;
Currency Settings	
Default Currency:	Euro a,- EUR
Significant Digits:	2 -
1000s Separator:	
Decimal Separator:	
Example:	à,-123.456.789,00
	Hilfe Zurļck Weiter

# Installation 6

Mit Weiter in den nächsten Dialog gehen.

Schritt 6: Einstellungen bestĤtigen	SUGARCRM
Bitte bestĤtigen Sie nachfolgende Einstellungen. Falls Sie Werte Ĥnd Schritt "zurļck". Ansonsten starten Sie nun die Installation mit "Weiter".	ern möchten, gehen Sie einen
Datenbank Konfiguration	
Datenbank-Name	sugarcrm (wollen wurde erstellt)
Datenbank-Benutzername	sugarcrm (wollen wurde erstellt)
Bestehende Sugar Datenbank-Tabellen wiederherstellen?	Nein
Datenbank mit Demo-Daten fÄ%len?	Nein
Priviligierter Datenbank-Benutzername	root
Seiten-Konfiguration	
URL der Sugar-Installation	http://172.16.110.22/sugarcrm
Update-Konfiguration	
Sugar Updates aktivieren?	Ja
Erweiterte Sicherheitseinstellungen	
Nutze ein individuelles Session-Verzeichnis fļr Sugar?	Nein
Nutze ein individuelles Log-Verzeichnis?	Nein
Unterstļtze die eigene Applikations-ID?	Nein
Locale Settings	
Default Date Format:	23.12.2006
Default Time Format:	23:00
Default Language:	US English
Default Name Format:	David Livingstone
Import/Export Character Set (Email, .csv, vCard, PDF, data import):	CP1252
Export (.csv) Delimiter:	i
Default Currency:	Euro
Currency Symbol:	â,¬
Currency Code (ISO 4217):	EUR
1000s Separator:	
Decimal Separator:	4
	Hilfe Zurżck Weiter

Installation 7





Dialoge erstellen

Seite 162	Schritt 7: Das Setup ausfļhren	SUGARCRM.
	Erstelle Sugar Konfigurationsdatei (config.php)	
	Erstelle Suggar Appliaktions-Tabellen, Audit-Tabellen und Metadaten	
	Erstelle Datenbank sugarcrm ein localhosterfolgt	
	Erstelle ein(en) Datenbank-Benutzernamen und -PasswortErstelle ein(	en) Datenbank-Benutzernamen
	relationships erstelle / Beziehungsdaten meta erfolgt	
	acl_roles erstelle / Beziehungsdaten meta erfolgt	
	erstelle / Audit Tabelle / Beziehungsdaten meta erfolgt	
	contacts erstelle / Audit Tabelle / Beziehungsdaten meta erfolgt	
	accounts erstelle / Audit Tabelle / Beziehungsdaten meta erfolgt	
	opportunities erstelle / Audit i abelle / Beziehungsdaten meta erto	Igt
	notes erstelle / Beziehungsdaten meta erfolgt	
	email_templates erstelle / Beziehungsdaten meta erfolgt	
	emailman erstelle / Beziehungsdaten meta erfolgt	
	calls erstelle / Beziehungsdaten meta erfolgt amails erstelle (Beziehungsdaten meta erfolgt	
	meetings erstelle / Beziehungsdaten meta erfolgt	
	iframes erstelle / Beziehungsdaten meta erfolgt	
	tasks erstelle / Beziehungsdaten meta erfolgt	
	users erstelle / Beziehungsdaten meta erfolgt	
	currencies erstelle / Beziehungsdaten meta erfolgt tracker erstelle / Beziehungsdaten meta	

#### Installation 8

Mit Weiter in den nächsten Dialog gehen.

Sprachpaket im Parameter local file angeben: SugarCRM\_langpack\_ge\_ge\_ 4-5-1.zip

Dann:





Install anklicken





5.2.5

Seite 163

Schind S. D	ownload & In	stall Language F	Packs	SUGARCRM.
Sie können hie nÃ≈chsten Schr	er ein weiteres Spra ritt.	achpaket installieren. /	Andernfalls kommen Sie r	nit "Weiter" zum
Sprachpake	t			
■ <u>Install</u> ■ <u>Install</u>	from Sugar Exchan from local file	ge		
Please ensure	e that you have cur	l enabled.		米
Folgende Spra	ichpakte kĶnnen n	un installiert werden:		
Folgende Spra Name Version	ichpakte kĶnnen n Publikationsdatui	un installiert werden: n Nicht deinstallierba	r Beschreibung	
Folgende Spra Name Version	ichpakte kĶnnen n i Publikationsdatui	un installiert werden: n Nicht deinstallierba	r Beschreibung	Install
Folgende Spra Name Version	ichpakte kĶnnen n I Publikationsdatui de Sprachpakete ir	un installiert werden: n Nicht deinstallierba stalliert:	r Beschreibung	Install
Folgende Spra Name Version Es sind folgen Name Versi	ichpakte kĶnnen n i Publikationsdatui de Sprachpakete ir on Publikationsda	un installiert werden: n Nicht deinstallierba Istalliert: tum Nicht deinstallier	r Beschreibung bar Beschreibung	Install

# Installation 10

Schritt 10: Registration	I SUGARCRM.
Please take a moment to regist little bit about how your compan product for your business need our mailing list. We do not sell, parties.	er with SugarCRM by filling out the short form below. By letting us know a y plans to use SugarCRM, we can ensure we are always delivering the right s. If you are interested in receiving information about Sugar, please opt-in to rent, share, or otherwise distribute the information collected here to third
Ihr Name und Ihre E-Mail Adres anderen Felder sind optional, a an dritte Anbieter verkaufen, ver	se sind die einzigen Felder die fĂ¼r die Registration benĶtigt werden. Alle ber sehr hilfreich. Wir werden die Informationen, die wir hier sammeln, nicht mieten oder weiterleiten.
* Required field	
About You	
* How did you hear about us?	Colleague
* First name:	Klaus-Peter
* Last name:	Luttkus
* Company Name	EPOS GmbH
* Job Title	CEO/President/Gen Mgr
* Email:	luttkus@eposgmbh.com
* Country:	Germany
Your Implementation	
* Your company's industry?	Education
* How many employees?	under 100 employees
* What is your current CRM solution?	Other 🔽
* How many SugarCRM users?	, 1-4
Keep me informed with Su a confirmation email sent to ab	garBUZZ newsletter, product information and special offers (Please respond to ove address)
Please have a SugarCRM	representative contact me
	Send Registration
	Hilfe Ende

# Installation 11

Ergänzung 11/2009

Mit Ende Installation beenden.



Dialoge erstellen

# 5.2.5 Seite 164

#### 5.2.5.19.5.3 SugarCRM aufrufen

Nach erfolgreicher Installation kann SugarCRM unter folgender URL aufgerufen werden:

http://<ip-Adresse vom php-Server>/sugarcrm

🚰 eposi501.eposgmbh.com - Microsoft Inter	net Explorer		_ _ ×
Datei Bearbeiten Ansicht Favoriten Extra	s ?		
🔇 Zurück 🛛 🕤 💉 🛃 💋 🔎	Suchen 🤺 Favoriten 🥝 🔗 🌭 🖀 - 🛄 🐒	1 🚳	
Adresse 🕘 http://eposi501.eposhbint.local/sugard	rm/index.php?action=Login&module=Users	💌 🄁 Wechseln zu	Links 🎽 🈏 SnagIt 📑
Google G → Go ↔ 💭	RS 🔹 🧐 👻 🎒 😴 📩 Bookmarks 🕶 🔊 23 blocked	🏀 Check 👻 🔦 AutoLink 👻	»> () Settings▼
	My Account Training Employees Logout About		SUGAR SUITE.
	Welcome to		
	SUGAR OPEN SOURCE		
	Please enter your username and password.		
	User Name		
	Login Votions		
	Server response time: 0.59 seconds.		
	© 2004-2007 SugarCRM Inc. All Rights Reserved.		
e) Fertig		🔹 🚺 📸 Interr	net 🧳

# Anmeldung

Melden Sie sich jetzt mit dem im Bild Installation 4 erstellten Benutzer an.





# 5.2.5.19.5.4 Anfangskonfiguration

Nach der ersten Anmeldung müssen Sie eine Anfangskonfiguration durchführen. Dazu gehören:

- Festlegen der Zeitzone
- Festlegen der Dialogsprache
- SugarCRM-Benutzer anlegen

Nach der Anfangskonfiguration erhalten die Benutzer folgenden Anmeldungsdialog.

Bitte geben Sie Passwort ein.	Ihren Benutzernan	nen und das
Benutzername	luttkus	
Passwort		
	Anmelden	
Theme [	Sugar	•
Sprache	Deutsch US English	<b>-</b>
	Deutsch	

Sprache wählen







**Dialoge erstellen** 

Nach der erfolgreichen Anmeldung präsentiert sich SugarCRM so: 5.2.5 🐸 eposi501.eposgmbh.com - Mozilla Firefox Seite 166 \_ 🗆 🗙  $\underline{D}atei \quad \underline{B}earbeiten \quad \underline{A}nsicht \quad \underline{C}hronik \quad \underline{L}esezeichen \quad \underline{Ex}tras \quad \underline{H}ilfe$ 🚑 🔹 📄 👻 🚱 🏠 📦 http://eposi501.eposhbint.local/sugarcrm/index.php?module=Home&action=inde • 🕨 💽 • Google 🗅 PHP 🗋 EPOS Intern 📋 www.midrangemaga... 🗋 Javascript 🄌 SELFHTML 8.1.1 (HT... 📋 #php/QuakeNet Tut... 🏧 ISeries Access für W.. 📦 eposi501.eposgmbh.com 🛛 🚨 📃 Wilkommen Klaus-Peter Mein Konto Training Mitarbeiter Admin Abmelden Über 🗌 🗌 🗖 🗛 🗛 SUGAR SUITE. ٠ EPOS Gmbh DV-Consult & Training 🚨 Suchen 🛛 🛛 🖓 Go Home Firmen Kontakte E-Mails Dokumente Zuletzt angesehen: 88 Klaus-Peter Luttkus SCHNELLMENUE Dashlet hinzufügen ? Hilfe R Neuer Kontakt COX JOTPAD Doppelklicken unterhalb für die Bearbeitung MEINE ANRUFE 002× 🐑 Eingabe Visitenkarte 
 Start
 Zurück: (0 - 0 von 0) Weiter \ Ende \N)
 Order manegement systems. There is no limit on creativity... You can outcomer your honepage to ft your needs. \* Add new Dashets by clicking on \*Add Dashets' in the upper left correr. \* Customize each Dashet by clicking on the Contrave to mit to comer of each Dashet to bashet by clicking on the Contrave to mit the Sugar Villa.

 Ørder manegement systems. There is no limit on creativity... You can outcome of each Dashet by clicking on the Contrave to mit to comer of each Dashet by clicking on the Contrave to mit the Sugar Villa.

 Ørder wanegement systems. There is no limit on creativity... You can outcome of each Dashet by clicking on the Contrave to mit to comer of each Dashet by clicking on the Contrave to mit the Sugar Villa.
 🖄 Neue Firma Beenden Betreff - Dauer Startdatum - Beginn - Bestätigen? 🔛 Neue E-Mail NEUER KONTAKT MEINE MEETINGS Vorname: MEINE OFFENEN FÄLLE ×550 Dauer Startdatum - Beginn - Bestätigen? Nachname: \* Beenden Betreff 👄 🖉 Start 🗹 Zurück (0 - 0 von 0) Weiter 🖻 Ende 🕅 
 CODX
 Fehler
 Titel
 Priorität
 Status
 O
 Telefon: MEINE INTERESSENTEN E-Mail: LISTE DER TOP-VERKAUFSCHANCEN 200x Name 🔶 Bürotelefon 🌣 Erstellt 🌣 ✓ Start ✓ Zurück (0 - 0 von 0) Weiter ▷ Ende ℕ Speichern ØX Verkaufschance llame ⇔ Betrag ⇔ Abschluss erwartet am ↔

Diagramm aktualisieren Pipeline total is €0,00K

MEINE ACCOUNTS

002X

|" Start ✓ Zurück (0 - 0 von 0) Weiter ▷ Ende ℕ

Kunde 👄 Telefon 👄 Angelegt am 👄



Übertrage Daten von eposi501.eposhbint.local...

MEINE PIPELINE

Prospecting

Qualification

Needs Analysis







# Das Kapitel wird überarbeitet.





5.3 Seite 2







# Das Kapitel wird überarbeitet.





5.4 Seite 2



# **2**

# 5.5 Web-Anwendungen auf dem HTTP-Server

Noch immer stellen sich IBM i-Betreiber nur zu oft die Frage, wie sie den Weg von der 5250-Oberfläche zu einer modernen Anwendungsoberfläche optimal meistern können. Auch wenn die Greenscreen-Oberfläche insbesondere bei Masseneingaben in Sachen Geschwindigkeit ihres Gleichen sucht, scheint deren Zeit doch abgelaufen. Schließlich bestimmen grafische Oberflächen längst den Alltag.

"Moderne Anwendungen" – darunter verstehen zu viele Unternehmen und Entscheider leider auch heute noch lediglich moderne Oberflächen, die auf alten Säulen stehen. Nur das Front-End einer Optimierung zu unterziehen, muss zwar nicht verkehrt sein – so lange die Säulen einer Anwendung so gebaut sind, dass sie auch zeitgemäßen Anforderungen Stand halten – doch letztlich ist es eine Konzeptionsfrage, ob und wie man zu modernen Anwendungen gelangt. Dabei sollten alle Anwendungsebenen – von der Datenbank über die Geschäftslogik bis hin zur Präsentationsschicht – eingehend analysiert und entsprechend angepasst werden.

Dass mit dem System IBM i eine Menge Werkzeuge und Anwendungen verfügbar sind, die als Grundlage für die Umsetzung einer Vielzahl von Anforderungen genutzt werden können, ist vielen Anwendern leider nicht bewusst, was auch damit zu tun hat, dass IBM hier nicht gerade "offensiv informiert".

In den vorhergehenden Abschnitten habe ich Sie bereits mit einigen Techniken vertraut gemacht, mit denen Anwendungsmodernisierung betrieben werden kann, Lösungen, die als Ersatz oder Erweiterung der 5250-Oberflächen genutzt werden können. An dieser Stelle möchte ich Ihnen nun ein weiteres Mosaiksteinchen genauer beschreiben, den Webserver, außerdem einige Anwendungen mit viel Potenzial, die als Baustein im Modernisierungsplan genutzt werden können.

WebFacing oder HATS (WDHT) sind Anwendungen der IBM, mit denen Entwickler, grafische Oberflächen für bestehende Anwendungen erzeugen können. Allerdings muss dazu vom IBM i-Server auch noch eine Ausführungsgrundlage bereitgestellt werden, ein Webserver, der für IBM i in Form des HTTP-Servers powered by Apache angeboten wird.

Dieser Server ist nicht nur die Grundlage für die Ausführung von Webanwendungen, die auf WebFacing, HATS oder WDHT basieren, sondern beispielsweise auch für Administrationsaufgaben notwendig, die über die grüne Oberfläche hinausgehen. Zudem bildet der Webserver die Basis für die Ausführung einfacher Webanwendungen, die als CGI-Anwendungen zum Einsatz kommen können.





Konfiguration eines HTTP-Servers auf dem System IBM i, die als Grundlagen für unterschiedliche Weboberflächen in Verbindung mit klassischen IBM i-Anwendungen nutzbar sind.





# 5.5.1 Seite 1

# 5.5.1 HTTP-Server-Konfiguration

Die Grundlage für die Ausführung einer Vielzahl grafischer Anwendungen bilden Webserver. Ein solcher Webserver ist im Standardlieferumfang eines jeden IBM i enthalten: der HTTP-Server powered by Apache. Dieses Lizenzprogramm (57xx-DG1) ist die Basis für die Darstellung von webbasierten Anwendungen bzw. Oberflächen. Bevor dieser Webserver genutzt werden kann, sind einige Einrichtungsschritte erforderlich. Die dazu notwendigen Schritte variieren in Abhängigkeit vom eingesetzten Betriebssystem. Nachfolgend finden Sie die Schritte, die für die Konfiguration eines Webservers unter der Betriebssystemversion 6.1 des IBM i erforderlich sind:

- 1. Starten des HTTP-Admin Bereichs unter TCPIP STRTCPSVR SERVER(\*http) HTTPSVR(\*ADMIN)
- 2. Starten des Browsers (in unserem Beispiel des Internet Explorers)
- Angabe der Start-URL f
  ür die HTTP-Server-Verwaltung (Hostname und der Admin-Port 2001)
   Zum Beispiel http://aszebis:2001 (f
  ür eine ungesicherte Verbindung)
   bzw. http://aszebis:2010 (f
  ür eine gesicherte Verbindung)

Der Einstieg in den Verwaltungsbereich der Webserver erfolgt über die Startseite des "IBM Systems Director Navigator for i5/OS". Hier melden wir uns mit einem IBM i-Benutzerprofil an, das über ausreichende Berechtigungen für die Anpassungen der Webanwendungen verfügt.

**Ein ITP Handbuch** 

Web-I

# Web-Programmierung

HTTP-Server-Konfiguration



HTTP-Admin-Startseite

Nach Angabe des IBM i-Benutzernamens und des Kennworts gelangen wir auf die Startseite der Administration.

Im linken Navigationsbereich befindet sich u. a. der Eintrag "Internetkonfiguration". Dahinter verbergen sich die Funktionen, mit denen der Webserver des IBM i verwaltet werden kann.



HTTP-Server-Konfiguration



5.5.1 Seite 3

🖉 - IBM Systems Director Navigator for	i5/0S - Windows Internet Explorer					
💽 🗢 🛃 https://aszebis:2005/ibm/con	isole/login.do?action=secure	<b>P</b> -				
Datei Bearbeiten Ansicht Eavoriten Extras 2						
🖕 Favoriten 🛛 👍 🍘 Vorgeschlagene Sites 🕶 🖉 Web Slice-Katalog 🔹						
🥖 - IBM Systems Director Navigator for i5/05	🛐 🔹 🔝 👘 🖾 🚍 🔹 Seite 🗸 Sicherheit 👻 Extras 🗸 🌘	<b>0</b> - "				
IBM Systems Director Navigator for i5/OS*	Willkommen zeig Hilfe   Abmelden	BM.				
Anzeigen: Alle Tasks 💌	Internetkon × Select Action	~				
<ul><li>Willkommen</li><li>Eigene Startseiten</li></ul>						
🗄 i5/OS-Verwaltung	Internetkonfigurationen - Aszebis ? -					
<ul> <li>System</li> <li>Basisoperationen</li> <li>Ablaufsteuerung</li> </ul>	i5/OS-Internetkonfigurationen ermöglichen das Ausführen von Internetkonfigurationstasks.					
Kondusteelening     Kondusteelening     Kondusteelening     Netzwerk     Verwaltung integrierter Server     Sicherheit     Benutzer und Gruppen	Ermöglicht die Verwaltung und Konfiguration von HTTP- und Anwendungsservern.					
	Ermöglicht das Hinzufügen von 15/OS-Verwaltungstasks zu Ihren Webanwendungen.					
<ul> <li>Journalverwaltung</li> <li>Leistung</li> </ul>	Ermöglicht die Erstellung, Verteilung und Verwaltung digitaler Zertifikate.					
Datelsysteme     Internetkonfigurationen     Backup; Nacovery and Media     Services     Einstellungen	Ermöglicht die Konfiguration von IBM IPP Server.					
	Ermöglicht die Verwaltung des Web-based Help Server.					
	<u>Assistent für Internet-Setup</u> Ermöglicht die Verwendung des Assistenten für Internet-Setup zum Konfigurieren Ihrer Internetumgebung. Schließen					
		_				
	States Intranet 🦓 🗸 🔍 🔩 100%	• .:				

 $\ Internet konfiguration$ 

Nach einem Klick auf den Assistenten für Internet-Setup klicken Sie in der nächsten Anzeige auf das Symbol "i5/OS-Taskseite".

IBM Systems Director Navigator for i5/OS - Wi	ndows Internet Explorer	
🕒 🕞 🔻 🙋 https://aszebis.2005/ibm/console/secure/s		🖌 😵 Zertifikatfehler 🦸 🗙 🔽 Bing
_atei Bearbeiten Ansicht Eavoriten E⊻tras ?		
🌪 Favoriten   🖕 🏉 Vorgeschlagene Sites 🔹 🔊 Web S	ce-Katalog •	
6 - IBM Systems Director Navigator for 15/05		🦄 👻 🗟 🕤 🚍 🦛 🗸 Seite - Sigherheit - Extras - 🔞
IBM Systems Director Navigator for i5/OS*	Willkommen zeig	Hilfe   Abmelden
Anzeigen: Alle Tasks 💌		Select Action
Willkommen		
Eigene Startseiten		
<ul> <li>System</li> <li>System</li> <li>Ablacksexenop</li> <li>Ablacksexenop</li> <li>Konflypration und Service</li> <li>Netework</li> <li>Vernaltang integrister Server</li> <li>Sicherhalt</li> <li>Benutzer vin Groppen</li> <li>Databasilen</li> <li>Jaurah Annotatug</li> <li>Lasting</li> <li< td=""><td>. 105-Taskraita</td><td></td></li<></ul>	. 105-Taskraita	
		📢 Lokales Intranet 🛛 🛷 • 🔍 100%

*i5/OS-Taskseite* 





HTTP-Server-Konfiguration

5.5.1 Seite 4 Sie werden erneut zur Eingabe eines Benutzers und des zugehörigen Kennworts aufgefordert:

Verbindung her	stellen mit aszebis 🛛 🛛 🛛 🛛	
Der Server "aszeb Benutzernamen ur Warnung: Dieser S Benutzernamen ur (Basisauthentifizie	is" an "HTTP Admin" erfordert einen id ein Kennwort. jerver fordert das Senden von id Kennwort auf unsichere Art an rung ohne eine sichere Verbindung).	
Benutzername:	🔮 zeig 🛛 👻	
Kennwort:	•••••	
	Kennwort speichern	
	OK Abbrechen	Admin-Anmel

Nun gelangen wir in den Verwaltungsbereich des Webservers.

#### Anmerkung

Durch die Eingabe der URL http://Servername:2001/HTTPAdmin gelangt man direkt in den Verwaltungsbereich des Webservers.



HTTP-Server-Konfiguration



Seite 5

Sollten Sie für die Ausführung der Webanwendungen noch nicht über einen Webserver verfügen, dann muss dieser zunächst eingerichtet werden. Existiert bereits ein Webserver und kann er für die neuen Webanwendungen genutzt werden, dann können Sie die folgenden Einrichtungsschritte überspringen und den gewünschten Webserver direkt selektieren, um dort neue Anwendungen zu installieren.



Neuen Server erstellen

Ergänzung 19/2011

Existiert noch kein Webserver oder kann/soll ein bereits existierender Webserver nicht genutzt werden, dann kann ein neuer Webserver durch einen Klick auf den Navigationseintrag "HTTP-Server erstellen" angelegt werden.

Auf dem IBM i können wir eine beliebige Anzahl Webserver konfigurieren. Jeder dieser Webserver verfügt über einen eindeutigen Namen, der auch im IFS als Konfigurationsbasis gespeichert wird.



HTTP-Server-Konfiguration

5.5.1 Seite 6 Für die Neuanlage eines Webservers vergeben wir einen möglichst sprechenden Namen. Der von IBM vorgeschlagene Name kann natürlich überschrieben werden!

HTTP-Server-Administration aut	f ASZEBIS - Windows I	nternet Explorer				
() v 🖉 http://aszebis:2001/H1	TPAdmin	V (4) 🗶 🔽 Bing				
Datei Bearbeiten Ansicht Eavoriten	E <u>x</u> tras <u>?</u>					
🚖 Favoriten 🛛 👍 🏉 Vorgeschlagene :	👷 Favoriten 🛛 🎪 🏉 Vorgeschlagene Sites 👻 🔊 Web Site-Katalog 👻					
HTTP-Server-Administration auf ASZEBI	s	🟠 🔻 🔂 👘 🖾 Sejte + Sigherheit + Extyras + 📦 + 🎽				
IBM Web Administration for i (O) WebSphere.						
✓ Allgemeine Tasks und Assistenten 2 Web Services Server erstellen	HTTP-Server er	stellen				
한 HTTP-Server erstellen 한 Anwendungsserver erstellen 한 WebSphere Portal erstellen	Willkommen beim Ass HTTP-Servers (power	istenten für die Erstellung eines neuen HTTP-Servers. Dieser Assistent hilft Ihnen beim Konfigurieren und Erstellen eines neuen d by Apache).				
	Sie müssen einen Na	men für den neuen Server angeben. Dieser Name wird später zur Verwaltung des Servers verwendet.				
	Welchen Namen soll	fer neue Server erhalten?				
	Server-Name:	WEBSERVER				
	Server-Beschreibung:	Demo WebServer				
	Klicken Sie auf Weiter, um fortzufahren, oder auf Abbrechen, um den Vorgang jederzeit abzubrechen.					
	Zurück Weiter	Abbrechen				
Fartia						
reruy		Suckales Intranet				

Neuen Server-Namen vergeben

Die Einstellungen des Webservers werden auf dem IBM i im IFS gespeichert. Auf der Folgeseite des Konfigurationsassistenten geben wir den Namen des IFS-Verzeichnisses für die Webserver-Konfiguration an. Dieses Verzeichnis befindet sich im integrierten Dateisystem der i5 (IFS). Außer den Konfigurationsdaten werden in Unterverzeichnissen neben Protokollen auch die Webseiten gespeichert, die später für die Anwender eingesetzt werden. Diese Webseiten sind die Benutzerschnittstellen für die Anwender, welche die grafischen Oberflächen zur Dateneingabe und Datenausgabe verwenden. Dieses wird voreingestellt, kann aber, wie der Name des Servers auch, individuell angepasst werden.


HTTP-Server-Konfiguration



Seite 7

In der folgenden Anzeige definieren wir, welcher Pfad für die Speicherung des HTTP-Servers verwendet werden soll. IBM schlägt als Einstellung grundsätzlich "/www/servername" vor.

HTTP-Server-Administration aut	f ASZEBIS - Windows Internet Explorer		
	TTPAdmin	💌 🗲 🗙 💽 Bing	P •
<u>D</u> atei <u>B</u> earbeiten <u>A</u> nsicht <u>F</u> avoriten	E <u>x</u> tras ?		
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene :	Sites 👻 🙋 Web Slice-Katalog 👻		
HTTP-Server-Administration auf ASZEB	15	🚹 🔻 🖾 🗉 🖷	) 🔹 Seite 🕶 Sigherheit 🔹 Extras 🕶 🔞 👻 🎽
IBM Web Administration for i Konfiguration Verwaltung   Erw	reitert   Zugehörige Links		WebSphere.
✓ Allgemeine Tasks und Assistenten <sup>1</sup> 2 Web Services Server erstellen	HTTP-Server erstellen		
인 HTTP-Server erstellen 인 Anwendungsserver erstellen 인 WebSphere Portal erstellen	Die Server-Root ist das Basisverzeichnis für Ihren Serve Konfigurationsdaten. Unterstützte Dateisysteme für die S	r. In diesem Verzeichnis erstellt der Assistent Unterverze Server-Root sind root und QOpenSys.	ichnisse für Ihre Protokolle und
	Welches Verzeichnis soll für die Server-Root Ihres neue	n Servers verwendet werden?	
	Server-Root: //www/webserver	Durchsuchen	
	Anmerkung: Ist das Server-Root-Verzeichnis nicht vorha	nden, wird es vom Assistenten erstellt.	
	Zurück Weiter Abbrechen		
			V
		Schales Intranet	🖓 🕶 🔍 100% 👻 📑

Server-Root

Nachdem die Schaltfläche "Weiter" angeklickt wurde, müssen wir ein weiteres Verzeichnis angeben – nämlich das sogenannte Dokument-Root. Dabei handelt es sich um das Stammverzeichnis, in dem die Serverdokumente bereitgestellt werden. Serverdokumente sind nichts anderes als die Webseiten, die wir mit dem HTTP-Server verwenden wollen. In dieses Verzeichnis wird während der Serverkonfiguration z. B. der Eintrag "index.html" gestellt. Dabei handelt es sich um das Startdokument für den Server.



HTTP-Server-Konfiguration

5.5.1	
Seite	8

	TPAdmin	💌 🖘 🔀 Bing	۶
atei <u>B</u> earbeiten <u>A</u> nsicht <u>F</u> avoriten	E <u>x</u> tras <u>?</u>		
Favoriten 🛛 👍 🏉 Vorgeschlagene	Sites 👻 🙋 Web Slice-Katalog 👻		
HTTP-Server-Administration auf ASZEB	5	🟠 🔹 🖾 👘 🔹 Seite - Sighe	heit 🔹 Extras 🔹 🔞 🕶
BM Web Administration for i			hSnhere
Konfiguration Verwaltung   Erw	eitert   Zugehörige Links		
Allgemeine Tasks und Assistenten	HTTP-Server erstellen		
UWeb Services Server erstellen UHTTP-Server erstellen			
Anwendungsserver erstellen	Die Dokument-Root ist das Basisverzeichnis, in dem Do	okumente durch Ihren Server bereitgestellt werden.	
a webophere i ortal erstelleri	Welches Verzeichnis soll für die Dokument-Root für Ihre	en neuen Server verwendet werden?	
	Dokument-Root: /www/webserver/htdocs	Durchsuchen	
	Anmerkung: Ist das Dokument-Root-Verzeichnis nicht v	orhanden, wird es vom Assistenten erstellt.	
	Zurück Weiter Abbrechen		
	Zurück) Weiter Abbrechen		
	Zurück Weiter Abbrechen		
	Zurück Weiter Abbrechen		

#### Dokument-Root

Auch hier schlägt das System wieder einen Namen vor. Wie man unschwer erkennt, besteht er wieder aus dem Root-Verzeichnis, gefolgt von dem Eintrag "htdocs". Diese Einstellungen übernehmen wir und setzen die Konfiguration mit einem Klick auf "Weiter" fort.

Die Kommunikation über die Weboberfläche erfolgt mittels TCPIP. Für die Anwendungen kann ein definierter Port genutzt werden, den wir zusammen mit der/den möglichen TCPIP-Adressen in der nächsten Anzeige des Konfigurationsassistenten angeben.



**E** Kapitel

Seite 9

HTTP-Server-Konfiguration

HTTP-Server-Administration auf A	SZEBIS - Windows Internet Explorer		
COO - E http://aszebis:2001/HTTF	Admin	🖌 🛃 🔀 Bing	P •
Datei Bearbeiten Ansicht Eavoriten	Extras ?		
👷 Favoriten 🛛 🚖 🏈 Vorgeschlagene Site	es 🔹 🙋 Web Slice-Katalog 🔹		
HTTP-Server-Administration auf ASZEBIS		🛐 👻 🔝 🝸 💼 👻 Sejte 👻 Sigherheit 👻	Extras 🔹 🔞 🕶 🦈
IBM Web Administration for i Konfiguration Verwaltung   Erweit	ert   Zugehörige Links	WebSph	ere. IBM
Algemeine Tasks und Assistenten Web Server erstellen Web Server erstellen Web Sphere Portal erstellen WebSphere Portal erstellen	HTTP-Server erstellen Der Server kann Anforderungen an bestimmten IP-Adress An welcher IP-Adresse und an welchem TCP-Port soll ih IP-Adresse: Mile IP-Adressen ♥ Port: 80 Anmerkung: Die meisten Browser stellen Anforderunger Zurück Weiter Abbrechen	isen oder an allen IP-Adressen des Systems entgegennehmen. r neuer Server Anforderungen entgegennehmen? n standardmäßig an Port 80.	
			M

### Portinformationen

Wahlweise können alle auf der i5 konfigurierten IP-Adressen verwendet werden. Es lässt sich aber auch nur eine der auf der i5 verwendeten IP-Adressen angeben – damit kann die Administration und Überwachung auf diese eine IP-Adresse beschränkt werden. Sie lässt sich durch einen Klick auf das Erweiterungssymbol rechts neben dem Feld "IP-Adresse" auswählen.

IP-Adresse:	192.168.68.40 💌	[
Port:	Alle IP-Adressen 192.168.68.40	
	192.168.68.60	

## Spezielle IP-Adresse

In unserem Beispiel verwenden wir die IP-Adresse 192.168.68.40 des IBM i.

Neben der IP-Adresse ist die Angabe eines Ports erforderlich, der mit der Voreinstellung "80" übernommen werden kann. Dieser Port lässt sich auch individuell anpassen.

Bei dem Port "80" handelt es sich um einen allgemein gültigen Standard, der für die Kommunikation mit einem HTTP-Server verwendet wird. Deshalb sind einige Standardanwendungen oder Standardwerkzeuge (wie z.B. der WebSphere Application Server) auch auf diesen Port ausgelegt und müssen in der Regel nicht speziell angepasst werden.



HTTP-Server-Konfiguration

# 5.5.1 Seite 10

Sollte der Port an dieser Stelle geändert werden, so ist die gezielte Verwendung dieses Ports beim Zugriff (über einen Browser, eine Anwendung oder einen Applikations-Server) auf den Server erforderlich.

In unserem Beispiel verwenden wir bewusst den Standardport "80", denn diesen benötigen wir u.a. für die Standardkonfiguration eines WebSphere-Servers.

Nachdem die IP-Adresse und der Port ausgewählt wurden, klicken wir auf "Weiter" und setzen die Konfiguration fort.

In vielen Fällen besteht ein besonderes Interesse, über Zugriffe auf bestimmte Bereiche informiert zu werden. Für einen HTTP-Server können wir in der nächsten Anzeige festlegen, ob eine Zugriffsprotokollierung durchgeführt werden soll. Damit lassen sich Auswertungen sowohl über Zugriffe auf den Web-Server generell als auch auf spezielle Seiten des Web-Servers erstellen. Unter anderem werden die IP-Adressen aufgezeichnet, mit denen auf den Server zugegriffen wird.

Neben solchen Zugriffsprotokollen wird zusätzlich ein Fehlerprotokoll mitgeschrieben, in dem alle Ausnahmefälle aufgezeichnet werden. Dieses Fehlerprotokoll wird in jedem Fall erstellt – unabhängig von der Auswahl, ob ein Zugriffsprotokoll geführt werden soll.

Als Empfehlung gilt, das Zugriffsprotokoll erstellen zu lassen.

CHTTP-Server-Administration au	f ASZEBIS - Windows Internet Explorer
💽 🗢 🙋 http://aszebis:2001/H	TPAdmin 🔽 🛃 Ding 🔎 🕨
<u>D</u> atei <u>B</u> earbeiten <u>A</u> nsicht <u>F</u> avoriten	Extras ?
🖕 Favoriten 🛛 👍 🏉 Vorgeschlagene	Sites 🔻 🙋 Web Site-Katalog 💌
HTTP-Server-Administration auf ASZEB	15 🚵 - 🖸 👘 - Sejte - Sigherheit - Extras - 🕢 - 🎽
IBM Web Administration for i Konfiguration Verwaltung   Erw	reifert   Zugehörige Links
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> <li>HTTS-Server erstellen</li> <li>Anwendungsserver erstellen</li> <li>WebSphere Portal erstellen</li> </ul>	HTTP-Server erstellen Ihr Server kann Aktivitäten auf ihrer Web-Site über ein Zugriffsprotokoll autzeichnen. Das Zugriffsprotokoll enthält informationen über die Anforderungen, die an den Sverver gestellt werden. Diese Informationen können verwendet werden, um zu analysieren, welche Benutzer die Web-Site benutzen und wie viele Anforderungen innerhalb eines bestimmten Zeitraumes gestellt wurden. Soll ihr neuer Server ein Zugriffsprotokoll verwenden?:
	Si Lokales Intranet 🦓 - 🔍 100% - 💥

Protokollinformationen



HTTP-Server-Konfiguration



Seite 11

Da ein solches Zugriffsprotokoll je nach Art und Einsatz des Servers recht ansehnliche Ausmaße annehmen kann, sollte das Protokoll nicht über einen unbestimmten Zeitraum geführt werden. Deshalb wird bereits bei der Konfiguration des HTTP-Servers eine Verwaltungsoption angeboten, mit der definiert wird, in welchen Zeitabständen das Zugriffs- und das Fehlerprotokoll reorganisiert werden sollen.

IBM bietet an dieser Stelle zwar auch die Option "aufbewahren – nicht löschen" an – allerdings sollte von der Option in der Praxis kein Gebrauch gemacht werden – denn in diesem Fall müsste eine manuelle Reorganisation der Protokolle erfolgen, was einen zusätzlichen Aufwand bedeutet. Deshalb verwenden wir in unserem Beispiel auch die empfohlene Methode der automatischen Bereinigung. Diese belassen wir auf dem Vorschlagswert "7 Tage".



Aufbewahrung

Was haben wir nun alles angegeben? In der letzten Anzeige zur Konfiguration eines neuen HTTP-Servers präsentiert das System eine Zusammenfassung aller getroffenen Auswahlen, um sie letztmalig zu kontrollieren. Hier besteht noch die Möglichkeit, durch die "Zurück"-Schaltfläche Änderungen an den getroffenen Auswahlen vorzunehmen.

IBM POWER Systems i Softwareentwicklung und -modernisierung



HTTP-Server-Konfiguration

5.5.1 Seite 12 Sind die getroffenen Auswahlen genau das, was wir wollen, so erstellen wir nun mit einem Klick auf die Schaltfläche "Fertigstellen" unseren neuen HTTP-Server.

HTTP-Server-Administration auf	ASZEBIS - Windows Internet Explorer		
	TPAdmin	💌 👉 🗙 📴 Bing	P -
Datei Bearbeiten Ansicht Favoriten	E <u>x</u> tras ?		
🚖 Favoriten 🛛 👍 🔏 Vorgeschlagene S	sites 🔻 🙋 Web Slice-Katalog 👻		
HTTP-Server-Administration auf ASZEBI	5	🦄 🔹 🗟 🔹 🖃 🖶 🔹 Sejte 🕶 Sigherheit 🗸	Extras 🔹 🔞 🔹 🎽
IBM Web Administration for i Konfiguration Verwaltung   Erw	eitert   Zugehörige Links	() WebSpher	a IBM
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> <li>→ HTTP-Server erstellen</li> <li>→ Anwendungsserver erstellen</li> <li>&gt;&gt; WebSphere Portal erstellen</li> <li>&gt;&gt;&gt; WebSphere Portal erstellen</li> </ul>	HTTP-Server erstellen       Server-Beschreibung: Dem OVebServer       Server-Beschreibung: Dem OVebServer       Dokument-Root:     //www.webserver       Dokument-Root:     //www.webserver/ldocs       IP-Adresse     Alle IP-Adressen       Port:     80       Protokollverzeichnis:     //www.webserver/logs       Zugriffsprotokolidatei:     erorc_log       Protokollverwaltung:     7 Tagen		
		😏 Lokales Intranet 🥠 🗸	🔍 100% 🔻 💡

Zusammenfassung der Installationsangaben

Nachdem die Konfiguration beendet wurde, wechselt die Browseranzeige in den Verwaltungsbereich des soeben konfigurierten HTTP-Servers.



*Startseite des HTTP-Servers* 



HTTP-Server-Konfiguration

Damit haben wir die Basiskonfiguration unseres ersten HTTP-Servers auf der i5 abgeschlossen.

Nachdem wir einen HTTP-Server installiert und auch konfiguriert haben, stellt sich natürlich die Frage, was wir alles mit einem solchen Server machen können. Neben dem Einsatz in Verbindung mit einem Applikationsserver lässt sich ein solcher Webserver z. B. auch einsetzen, um einfache Anwendungen auf Basis eines Browsers auszuführen. Dabei kann auch auf den Inhalt der Datenbank der i5, die DB2/UDB, zugegriffen und sogar deren Inhalt verändert werden.

Da der Webserver nun für den allgemeinen Einsatz vorbereitet ist, kann er beispielsweise als Grundlage für die Präsentation von Webanwendungen zum Einsatz kommen. Webanwendungen können mithin so ausgelegt werden, dass sie die 5250-Ausgabe ersetzen und die bestehenden RPG- oder Cobol-Anwendungen des IBM i nutzen.

Einfache Anwendungen können – wie gesagt – ohne einen Anwendungsserver direkt auf dem Webserver ausgeführt werden. Diese Anwendungen sind in Bezug auf die Architektur und die Anforderungen an die Laufzeitumgebung genügsam. Damit solche Anwendungen aber auch genutzt werden können, bedarf es noch einiger weniger Einstellungsanpassungen, die im Folgenden erklärt werden. Dazu wechseln wir in den Verwaltungsbereich des zuvor konfigurierten Webservers.

Die folgende Abbildung zeigt eine Übersicht der konfigurierten Server. Hier können Sie bei Bedarf auch zwischen den verschiedenen Webservern wechseln.

## ACHTUNG!

Verändern Sie keine der von IBM ausgelieferten Server. Diese Server dienen unter Umständen wichtigen Systemfunktionen als Grundlage!

HTTP-Server-Konfiguration

# 5.5.1 Seite 14

Det ig besten gest E worden gest	HTTP-Server-Administration au	f ASZEBIS - Windows	Internet Explorer					
Deel Berberten grott Evroten Egtes 2 * Aventur Seiter Sature State Versioner See Versioner See State Stat	🔄 🕞 🗢 🙋 http://aszebis:2001/H	TTPAdmin					💌 🐓 🗙 🔽 Bing	P -
Image: Server American Server	Datei Bearbeiten Ansicht Eavoriten	E <u>x</u> tras <u>?</u>						
IPTP-Server-Advisedation and ASSERTS       Image: Section 2 Signature 1: Expose Image: Section 2 Signature 2 S	🚖 Favoriten 🛛 👍 🏉 Vorgeschlagene	Sites 🔹 🙋 Web Slice-Ka	stalog •					
BM Web Administration for L       Image: Constraint of the con	HTTP-Server-Administration auf ASZEB	IS					🏠 🔹 🖾 🚽 📾 🔹 Seite 🖌 Sigherhi	sit • Extras • 🔞 •
Mei Stell Mir/Scherei Pputcabel detet Weißsphere Folden Weißsphere Folden MeitTP-Server (weiße MeitTP-Server (weiße MeitTP-Server (weiße Server- Kaleborg) Meitten Server- Kaleborg Meitten Statum Sta	IBM Web Administration for i Konfiguration Verwaltung Erw	reitert   Zugehörige Lin	iks				() Web:	Sphere IBM
Stand vom 10.06.2011 13.03.52.  Stand vom 10.06.2011 13.03.52.  Stand vom 10.06.2011 13.03.52.  Stand vom 10.06.2011 13.03.52.  Adresse:Port Zugeordneter Anvendungsserver Beschreibung Additive Apache(2.2.1105) Gestantet *2001 Keine Von IBM gelieferter Beispieri-HTTP-Server (powered by Apache) MADET Apache(2.2.1105) Gestappt *2020 Keine OWHEELESS Apache(2.2.1105) Gestappt *303 Keine Demo WebServer WEBSERVER Apache(2.2.1105) Gestappt *11331 WQLW/77  Server-Starbarameter  Catualisieren Stafta Stoppen Neustart Details verwatten Löschen Umberennen	Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTP-Server erstellen     Anwendungsserver erstellen	Alle Server ver	walten 📀	ver				
ADAILIN       Apachel2.2.11 ((5)       © Gestanct       *2001       Keine       Von IBM geliefenter Beispiel-HTTP-Server (powerd by Apachel)         APACHEDET:       Apachel2.2.11 ((5)       © Gestoppt       *30       Keine       Von IBM geliefenter Beispiel-HTTP-Server (powerd by Apachel)         WADET       Apachel2.2.11 ((5)       © Gestoppt       *30       Keine         OWNRELESS       Apachel2.2.11 ((5)       © Gestoppt       *300       Keine         OWNRELESS       Apachel2.2.11 ((5)       © Gestoppt       *300       Keine         WIESEENVER       Apachel2.2.11 ((5)       © Gestoppt       *1133       WOLWI77         Servier-Stangtameter:       Image: Control of the service in the servic	Ø WebSphere Portal erstellen	Stand vom 19.06.201 Server 🛦	1 13:03:52. Version	Status	Adresse:Port	Zugeordneter Anwendungsserver	Beschreibung	
APACHEDFT Apache2.2.11 (I5)      Gestoppt *30 Keine Von IBM geliefenter Beispiel-HTTP-Sener (powered by Apache)     MWAPET Apache2.2.11 (I5)      Gestoppt *2020 Keine     JummerLess Apache2.2.11 (I5)      Gestoppt *30 Keine     JummerLess Apache2.2.11 (I5)      Gestoppt *30 Keine     JummerLess Apache2.2.11 (I5)      Gestoppt *30 Keine     JummerLess Apache2.2.11 (I5)      Gestoppt *11331 WOLW77  Sener-Starbparameter:     JummerLess Starbparameter:     JummerLess Starbparameter:     JummerLess Apache2.2.11 (I5)      Gestoppt *11331 WOLW77		O ADMIN	Apache/2.2.11 (i5)	Gestartet	*:2001	Keine	ADMIN-Server	
MADET       Apache/2.2.11 (6)       © Gestoppt       *2020       Keine         O		O APACHEDFT	Apache/2.2.11 (i5)	9 Gestoppt	*:80	Keine	Von IBM gelieferter Beispiel-HTTP-Server (powered by Apache)	
Outballsvervalten     Starten     Starten     Starten     Utballsvervalten     Ubenennen		IWADET	Apache/2.2.11 (i5)	9 Gestoppt	*:2020	Keine		
WEBSERVER     Apachel 2.2.11 (I5)     Gestantet     *.80     Keine     Demo WebServer       WOLW77     Apachel 2.2.11 (I5)     Gestanpt     *.11331     WOLW77   Server-Startparameter       Aktualisieren     Stanten     Stoppen     Neustant   Details verwaiten       Löschen     Umbenennen			Apache/2.2.11 (i5)	9 Gestoppt	*:9004	QIWIRELESS		
WQLWI77       Apache/2.2.11 (I5)       Gestoppt       *:11331       WQLWI77         Server-Startparameter       Image: Starten Startparameter       Image: Starten Startparameter       Image: Starten Startparameter         Aktualisieren       Starten Startparameter       Image: Starten Startparameter       Image: Startparameter         Details verwaiten       Löschen Umbenennen       Image: Startparameter       Image: Startparameter		WEBSERVER	Apache/2.2.11 (i5)	Gestartet	*:80	Keine	Demo WebServer	
Serier-Startoarianeter Starten Stoppen Neustant Aktualisieren Starten Stoppen Neustant Detalls verwaiten Löschen Umbenennen		O <u>WQLWI77</u>	Apache/2.2.11 (I5)	Gestoppt	*:11331	WQLWI77		
Attualisierem     Starlen)     Stoppen     Heustart       Details verwaiten     Löschen     Umbenennen		Server-Startparameter	er.			9		
		Aktualisieren Details verwalten	Starten Stoppen	Neustart				

Serverübersicht

Im Verwaltungsbereich des ausgewählten Webservers finden wir in der linken Navigationssicht u. a. den Eintrag "Dynamische Inhalte und CGI". Wird dieser Eintrag ausgewählt, gelangen wir in eine Anzeige, die in der folgenden Abbildung zu sehen ist:

CHTTP-Server-Administration auf	f ASZEBIS - Windows Internet Explorer
COO - E http://aszebis:2001/HT	TTPAdmin 🖌 🖌 🔁 Bing 🖉 🖓
Datei Bearbeiten Ansicht Eavoriten	E <u>x</u> tras ?
🚖 Favoriten 🛛 👍 🏉 Vorgeschlagene S	Sites 🔻 🖉 Web Silce-Katalog 👻
HTTP-Server-Administration auf ASZEBI	15 🔐 + 🔂 - 🗈 🖨 - Sejte - Sigherheit - Extras - 👔 - 🎽
IBM Web Administration for i Konfiguration Verwaltung Erw	reitert   Zugehörige Links
Alle Server HTTP-Server Applic	cation Server
🕒 Gestartet [ 🔀 🔲 💋 Serv	ver: WEBSERVER - Apache 💌 Server-Bereich: Globale Konfiguration 💌
▼ Allgemeine Tasks und Assistente	WEBSERVER > Dynamische Inhalte und CGI Dynamische Inhalte und CGI
<ul> <li>✓ HTTP-Tasks und -Assistenten</li> <li>☑ Verzeichnis dem Web hinzufüge</li> <li>▲ LDAP-Konfiguration</li> </ul>	Server-Systemstart Persistente CGI Erweitert Allgemeine Einstellungen Umgebungsvariablen Server-seitige Includes
▼ Server-Eigenschaften	Maximale Anzahl aktiver CGI-Jobs: 40 Maximale Anzahl der aktiven CGI-Jobs mit Threads: 40 Ausführung von CGI-Programmen zulassen: Aktiviert 💙 🕑
D Anforderungsverarbeitung D HTTP-Antworten D Inhaltseinstellungen D Verzeichnisverarbeitung	Anmerkung: Klicken Sie auf die Indexzunge Enveitert, um die CGI-Programme zu aktivieren, die In mehrfach-thread-fähigen Jobs ausgeführt werden sollen.
Sicherheit Dynamische Inhalte und CGI Protokomeren	OK Anwenden Abbrechen Voranzeige
D Proxy D Systemressourcen D Cache ✓	
Fertig	😏 Lokales Intranet 🦛 👻 100% 👻

#### CGI-Einstellungen

Für die Ausführung einfacher Anwendungen direkt auf dem Webserver aktivieren wir die Einstellung "Ausführung von CGI-Programmen zulassen".



HTTP-Server-Konfiguration

Weitere Sitzungseinstellungen erlauben die Festlegung von Werten, die die Sicherheit und die Performance beeinflussen.



Persistente CGI

Übernehmen Sie vorerst die Einstellungen für die persistenten CGIs.

Damit ist die Grundlage für die Ausführung einfacher Webanwendungen auf dem Webserver geschaffen.



apite

Kapitel

Web-Programmierung HTTP-Server-Konfiguration

5.5.1 Seite 16



# 5.5.2 CGI-Anwendungen

Wie wir bei der Installation des Webservers gesehen haben, besteht u.a. auch die Möglichkeit, bestimmte Einstellungen für den Bereich CGI vorzunehmen. CGI (Common Gateway Interface) stellt eigentlich eine standardisierte Schnittstelle für den Datenaustausch zwischen einem Webserver und einem Programm dar. Mit Hilfe dieser Technik sind wir in der Lage, per HTML auf die Daten der i5 zuzugreifen und sie in Webseiten einzubinden. In diesem Sinne ist CGI eine einfache Variante, Webseiten mit dynamischen Inhalten zu befüllen.

CGI ist die einfachste Methode, die Daten der DB2/UDB für die Verwendung in einem Browser zugänglich zu machen und wird uns im Bereich der Anwendungsentwicklung mit webgestützten Werkzeugen immer wieder begegnen. Deshalb wollen wir uns die Hintergründe und Strukturen dieser Standardschnittstelle einmal etwas näher anschauen.

CGI ist eine im Standard von i5/OS ausgelieferte Schnittstelle, die für die Kommunikation zwischen einem Webserver und dem i5/OS dient (CGI ist nicht spezifisch für i5/OS, sondern existiert auch für andere Plattformen!).

Die nachfolgende Abbildung zeigt schematisch den Einsatz von CGI in Verbindung mit i5/OS.

# CGI schematisch

Jeder Datenzugriff über einen Browser – also über eine HTML-Seite – führt dazu, dass CGI die Datenanfrage an den Datenbankserver sendet, in unserem Fall also an die i5 oder genauer gesagt: an das Betriebssystem i5/OS.







Seite 1

# 5.5.2 Seite 2

In der Grafik "Die CGI-Schnittstelle" ist die Kommunikation bidirektional dargestellt. Das bedeutet, dass die Kommunikation zwischen CGI und dem Datenbanksystem in beide Richtungen ausgeführt werden kann. Diese Funktionalität zeigt sich z. B. bei der Verwendung spezieller APIs, mit denen sich die Datenbereitstellung innerhalb eines Browsers aus RPG heraus vornehmen lässt. Damit lassen sich kleine Anwendungen realisieren, bei denen z. B. einmalige Datenabfragen getätigt werden. Die CGI-Schnittstellenanwendungen sind in der Regel einfache Anwendungen, die sich durch geringe Komplexität auszeichnen. Im Gegensatz dazu stehen komplexere Anwendungen, die beispielsweise eine Vielzahl unterschiedlicher Daten verarbeiten und mehrere Programme oder Module einbinden und ausführen.

Nehmen wir dazu ein klassisches Beispiel: eine einmalige Bestandsabfrage auf einen Artikel. Diese Bestandsabfrage kann wahlweise durch eine eigene Programmlogik erfolgen, die für den Zugriff über einen Browser den aktuellen Datenbestand aus dem ERP-System der i5 ausliest. Allerdings ist es in der Praxis – und gerade auch für Anfänger – eher schwierig, sich mit dem Thema so zu beschäftigen, dass sich neben dem Front-End "5250-Ersatz durch Browser" auch noch die gesamte Programmlogik eventuell in Java oder einer anderen modernen Entwicklungssprache ersetzen lässt. Also bietet sich die Verwendung des bekannten RPG mit einem CGI-API an, um den Datenbestand auf der i5 bereitzustellen und ihn dem Browser einmalig zur Ansicht zu übertragen.

Dem aufmerksamen Leser ist vielleicht der zweifache Gebrauch des Wortes "einmalig" im Absatz zuvor aufgefallen. Wenn der Einsatz von CGI völlig problemlos wäre, dann hätten Produkte wie WebSphere, WebFacing oder HATS keine wirkliche Existenzberechtigung, denn: Beim Einsatz von CGI-Schnittstellen lassen sich keine permanenten Verbindungen aufbauen. Das heißt, dass nur kurzeitig, nämlich für den Datenaustausch – bzw. die Datenanfrage – und ggf. zusätzlich nochmals für die Übertragung des Ergebnisses der Datenermittlung eine Verbindung zwischen dem HTTP-Server und der Datenquelle hergestellt wird. Die Verbindung bleibt also anders als beim Einsatz von 5250-Anwendungen nicht bestehen, sondern wird nach der Ausführung einer Transaktion wie z. B "Lagerbestand anzeigen" wieder abgebrochen, um dann nach Ermittlung des Lagerbestandes auf der i5 für die Rückantwort wieder aufgebaut zu werden.

Diese Einschränkung verhindert auch den breiten Einsatz von CGI-Anwendungen in komplexen Umgebungen. Der Aufbau einer Verbindung führt nämlich zu einer relativ starken Belastung der Systemressourcen. Darüber hinaus kann es bei bestimmten Konstellationen zu Problemen oder langen Wartezeiten kommen, die einen sofortigen Aufbau einer Verbindung verhindern – bei wirklich komplexen und wichtigen Anwendungen ein nicht zu vertretender Zustand.



Diesen Vorgang sehen wir in den nächsten beiden Abbildungen schematisch dargestellt.

Zunächst stellt ein Anwender eine Anforderung "Lagerbestand abfragen". Diese wird durch eine Browsereingabe initiiert.

Nach dem Auslösen der Anfrage über einen entsprechenden Mausklick oder eine andere Aktion, wird über die CGI-Schnittstelle die Anforderung an das Datenbanksystem übergeben.

Die Anforderung benötigt zunächst eine Verbindung zum externen System, die durch die CGI-Schnittstelle automatisch aufgebaut wird. Nach dem Verbindungsaufbau lassen sich dann die erforderlichen Befehle und Transaktionen ausführen, mit denen die gewünschten Informationen bereitgestellt werden können.



## Beispielanwendung CGI

Je nach Art und Komplexität der Anforderung kann die Ermittlung der Daten unterschiedlich lange dauern. Die Verbindung, die über die CGI-Schnittstelle zum Zweck der Datenanfrage aufgebaut wurde, wird automatisch wieder beendet. Damit die Ergebnisse dem Benutzer zurückübertragen werden können, muss also erneut eine Verbindung aufgebaut werden. Diesmal vom Datenbankserver (der i5) in Richtung des Browsers.





CGI-Anwendungen

# 5.5.2 Seite 4

Zwar ist PHP als Programmiersprache für die iSeries wenig verbreitet, aber vielleicht befindet sich in der Schar der Anwendungsentwickler doch der eine oder andere Programmierer, der über PHP-Kenntnisse verfügt. Anwendungen, die in PHP geschrieben sind, lassen sich mit Hilfe des HTTP-Servers nämlich auch auf der iSeries ausführen, wo diese Programme auf DB2/UDB-Daten zugreifen können.

Eine andere – leider auch viel zu wenig bekannte Programmiersprache, die mit einem HTTP-Server verwendet werden kann, ist Net.DATA – eine IBM-spezifische Sprache, die bereits seit einigen Jahren zur Verfügung gestellt wird. Leider ist diese Art der Anwendungsentwicklung wenig verbreitet – zu unrecht – aber Details zu Net.Data würden den Rahmen dieses Kapitels sprengen.

Allerdings ist der Net.DATA Ansatz nur für die Entwicklung einfacherer Webseiten bzw. kleinerer Webanwendungen gedacht, denn neben dem Vorteil relativ einfacher Anwendung und Realisierung ist mangelhafte Ausführungsgeschwindigkeit gegenüber Servlets oder auch PHP ein deutlicher Nachteil. Trotzdem stellt CGI eine ideale Möglichkeit dar, die ersten Schritte mit der Anbindung von i5-Daten ins Web zu wagen.

Bei CGI handelt es sich, vereinfacht gesagt, also auch um ein Programm, das von einem HTTP-Server aufgerufen werden kann und mit dem Informationen aus einem Datenbanksystem verarbeitet werden können. Das aufgerufene Programm kann in einem solchen Fall in einer beliebigen Sprache erstellt worden sein. Deshalb lassen sich im Zusammenhang mit der i5 z. B. auch RPG- oder COBOL-Programme mit CGI einsetzen. Das aufgerufene Programm liest beispielsweise Eingabefelder aus und führt eventuell erforderliche Berechnungen durch. Als Ergebnis können auch Daten aus der DB2/UDB zur Anzeige in einem Webbrowser gebracht werden.

Eines der am häufigsten eingesetzten Tools im Bereich von CGI mit i5 ist EASY400 bzw. CGIDEV2. Dabei handelt es sich um eine Ansammlung verschiedener Programme und Funktionen, mit deren Hilfe wir DB2/UDB-Daten in eine Webanwendung einbinden können. Dieses Tool wird von IBM kostenlos zum Download angeboten: http://www.easy400.net/easy400p/maindown.html.





CGI-Anwendungen



Seite 5



Download CGIDEV2

Auf dieser Webseite stehen neben dem CGIDEV2-Tool noch einige weitere sehr nützliche Tools zum Herunterladen bereit. Wir wollen uns an dieser Stelle aber auf das das CGIDEV2-Tool beschränken.

Auf der genannten Webseite stehen uns zwei verschiedene Versionen zur Verfügung. Die klassische Version, die ab OS/400 V5R1 eingesetzt werden kann, ist auch die Basis für unsere Beispiele. Laden Sie sich diese Version zunächst aus dem Internet herunter. Ich verzichte an dieser Stelle bewusst auf die Darstellung der einzelnen Schritte, die im Rahmen des Downloads ausgeführt werden müssen und beschränke mich auf die Beschreibung der Installation und den weiteren Einsatz des Tools.





CGI-Anwendungen

5.5.2 Seite 6 Der Download von CGIDEV2 erfolgt in Form einer Zipdatei, die zwei Einträge enthält: Eine Textdatei, die weitere Informationen über CGIDEV2 enthält und eine Sicherungsdatei mit den auf der i5 zu installierenden Komponenten.

🗐 Wi	inZip (Un	registrier	t) - cgidev2.	.zip			×
Datei	Aktionen	Optionen	Hilfe				
Č			1	<b>(</b>	<b>E</b>	B	
9	Veu	Öffnen	Favoriter	n Hinzufügen	Extrahieren	Verschlüsseln	A
Name						Тур	
🖬 cgia	dev2.savf					SAVF-Datei	
🛄 Rea	adme.txt					Readme-Datei	
							-0
<		1111					>
0 Datei	en (O Byte)	ausgewählt		2 Dateien (5.490k	(B) insgesamt	0	)

Inhalt der Zipdatei

Die Sicherungsdatei übertragen wir zunächst nach dem gewohnten Verfahren auf die i5 und speichern den Inhalt dort mit dem Befehl "RSTLIB SAVLIB(CGIDEV2) DEV(\*SAVF) SAVF(CGIDEV2)" zurück.

Um die Installation zu vervollständigen, befolgen Sie bitte die Anweisungen, die sich in der Datei README in der Bibliothek CGIDEV2 befinden:

	Mit Teildateien	arbeiten (mittels PDM)	SOEPRO
Datei Bibliothek .	. <u>README</u> . <u>CGIDEV2</u>	Listenanfang be	ei
Auswahl eingebe 2=Editieren 8=Beschreibung	n und Eingabetaste dr 3=Kop. 4=Lösch. 5=A anz. 9=Sich. 13=T	ücken. nzeigen 6=Drucker ext ändern 14=Umwand.	n 7=Umbenennen 15=Modul erst
Ausw. Teildatei CGIDEV2 CHANGES INSTALL	ArtTextHTMLCGIDEV2TXTChangesTXT	<u>readme HTML file 2005 log last changed 2006</u>	5-07-28 15:30 8-03-27 14:00:00
Parameter oder ===>	Befehl		Ende
F3=Verlassen F9=Auffinden	F4=Bedienerführung F10=Befehlseingabe	F5=Aktualisieren F23=Weitere Angaben	F6=Erstellen F24=Weitere Tasten

#### Installationsanleitung

Nach der Installation des Tools sollten Sie die Anleitungen befolgen. Damit werden u.a. die Voraussetzungen für einen sicheren und erfolgreichen Betrieb des Tools definiert.



Das Schöne an diesem Tool ist, dass man mit ihm die verschiedenen Beispiele für Webanwendungen direkt auf dem System ausführen kann. In der Regel besteht eine solche Webanwendung aus einem HTML-Bereich und einem dazugehörigen i5-Programm wie z. B. einem RPG-Programm. Beides ist im Lieferumfang von CGIDEV2 enthalten.

Bevor wir CGIDEV2 nutzen können, müssen einige Konfigurationseinstellungen vorgenommen werden. Ich verzichte an dieser Stelle bewusst auf das Abschreiben der Installationsanleitung, die im Internet unter der folgenden Adresse eingesehen werden kann: http://www.easy400.net/cgidev20/cgisetup.htm

Befolgen Sie die auf dieser Webseite enthaltene Installations- und Konfigurationsanleitung. Mit der Ausführung dieser Anleitung wird auf dem Webserver der i5 eine vollständige Anwendung bereitgestellt, die eine Vielzahl unterschiedlicher Beispielanwendungen bietet. Diese Beispiele können wir adaptieren und als Grundlage für eigene Anwendungen nutzen.

Geben Sie für den Aufruf der Webanwendung die folgende URL ein: http://ipadresse:port

Ersetzen Sie in der URL "ipadresse" mit dem Hostnamen bzw. der IP-Adresse der i5 und verwenden Sie den Port, der diesem Webserver zugeordnet wurde.

Das Ergebnis dieser Eingabe sollte dann die folgende Anzeige sein:

Intel Beschellen Anscht Fordran Extras ?         Charles Beschellen Anscht Fordran Extras ?         Charles Control (Vigdar/2)tact         Charles Control (Vigdar/2)tact         Data dema	Wechseh zu Links <sup>1</sup> 07-11 15:51 CEI
Construction of the series way for you to develop Web applications is to use the CGI technique in not easy by stell. However we have made it to easy that after a little reading you would at back and multiple accesses web programma is now just a your hand. As a matter of fact we teach, and provide tools, and structed on a service arowy any complexity from your code, while along a reader of the teach of teach of the teach of tea	Wechseln zu Unis <sup>1</sup> 07-11 15:51 CEI
	Wechseln zu Unks
It is not easy by theff. However, we have made it to easy that after a list reading your of fact we have made it to easy that after a list reading your of fact we have made it to easy that after a list reading your of fact we have made it to easy that after a list reading your of fact we have made it to easy that after a list reading your of the track, and provide tools, and arrive program and and on a service program and your your grown are used to define your estemal Web pager, such as a service program and your grogram.	07-11 15:51 CE:
Get Series to the WEB with your RPG  I ast update. May 23, 2005  If you know RPG, the saniest way for you to develop. Web applications is to use the CGI technique. The CGI achinque is not easy by itself. However we have made it to easy that after a latte reading you would at back, and mile, because web programming is now just aty your hand. As a matter of fact we teach, and provide tools, or a technique, back on a service perogram, that indices arowy any complexity from your code, while allowing you to define your. Web pages external to your program. Web pages external to your program. If you can be used to define your external Web pages, such as	
I lost updato Mark Mary Varia	2
If you know RPG, the easiest way for you to develop Web applications is to use the CGI technique. The CGI techniques is not easy by intell However we have made it to easy that after a linte reading you would site back. If provides a many better to define your mode and you be approxements in the develop and	uat's CGI?
Hind, xnu, vnai, yavascrupt This service programs is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce programs is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce programs is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce programs is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce programs is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. In this struce program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. Rothest program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. Rothest program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. Rothest program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. Rothest program is originated by Mel Rothman, IBM iSenes 400 Curtom Technology Center, Rochester (MD), U.S. Rothest program is originated to Version 2 display this sign <sup>2</sup> / <sub>2</sub> Check out <u>interprovedated</u> Check out <u>interprovedated</u>	
<ul> <li>Learning path</li> <li>Second Acroba Reader)</li> <l< td=""><td>rvice program. uccess the <i>rogram</i> with your PC Hs, maybe it's</td></l<></ul>	rvice program. uccess the <i>rogram</i> with your PC Hs, maybe it's

Die Startseite des CGIDEV2-Tools auf dem Webserver







# 5.5.2 Seite 8

Nicht dass Sie sich wundern, aber was wir jetzt sehen, ist eine Webseite, die auf der i5 ausgeführt wird. Es handelt sich dabei um den Willkommensbildschirm des CGIDEV2-Tools. Wie Sie sehen können, besteht diese Anzeige aus verschiedenen Teilbereichen, mit denen wir Zugang zu den unterschiedlichsten Funktionen haben.

Mit einem Klick auf den Bereich "Basic Demos" gelangen wir in den Bereich der Beispielprogramme. Das Schöne dabei ist, dass nicht nur die fertigen und lauffähigen Anwendungen von hier aus aufgerufen werden können, sondern dass wir sowohl den HTML-Code, der für diese Webseiten benötigt wird, als auch den RPG-Code von dieser Webseite aus aufrufen und einsehen können. Damit fällt der Zugang in diese neue Welt der Anwendungsentwicklung wirklich leicht.

Ohne dass wir uns mit jedem der einzelnen Beispiele im Detail beschäftigen können, möchte ich an dieser Stelle doch eine der in CGIDEV2 enthaltenen Beispielanwendungen im Detail behandeln.

Bei besagter kleiner Beispielanwendung können wir mit Hilfe von zwei Feldern Informationen eingeben, die dann bei einer erneuten Ausgabe der HTML-Seite enthalten sind. Des Weiteren wird in diesem Programm ein Zugriff auf den Inhalt von Systemdaten realisiert. Mit einem eigenen Programm, das innerhalb der Logik dieser Webanwendung ausgeführt wird, erfolgt ein Zugriff auf den Releasestand der Maschine, der als Rückgabewert auf der Webseite ausgegeben wird.

Klicken Sie in der Begrüßungsseite auf den Eintrag "Basic Demos", wie es die folgende Abbildung zeigt:



Starten der Basic Demos





In diesen "Basic Demos" ist u.a. das Programm "Hello" enthalten, das wir mit einem Klick auf das gelbe Symbol "Go" starten können. Schauen Sie sich dazu auch die folgende Abbildung an.



Starten des Beispielprogramms "Hello"

Dieses einfache Programm erlaubt die Eingabe von zwei Variablen "First Name" und "Last Name". Zwar werden diese Variablen nicht in einer Datei gespeichert, stehen aber im zweiten Schritt der HTML-Verarbeitung wieder als Ausgabewerte zur Verfügung. In diesen Feldern können Sie einen beliebigen Wert eingeben.





CGI-Anwendungen

🛛 🜍 Zurück 🔹 🐑 👻 🛃 😭 🔎 Suchen 🌟 Favoriten 🚱 🖾 🗞 🔯 🔹 🛄 💱 🎇 🦓									
Adresse 🕘 http://soepro:8014/cgidev2o/hello.htm									
Test your http server functionality for CGIs									
This html is intended to test your http server for CGI support. Enter your first and last names below, and press the <b>Send</b> button.									
Your first name Willi									
Your last name Winzig									
Send									
Then, take a look at what happens. If									
1. program /cgidev2p/hello1.pgm responds, providing									
o your first name									
o your last name									
then your http server is all right.									
<ol> <li>program /cgidev2p/hello1.pgm responds, but does not provide one of the following o your first name</li> </ol>									
o your last name									
then your http server is missing some PTsF for the QtmhCvtDb API. Check page Tips for installation on your AS/400 to get a list of minimum PTF requirements.									

### Das Beispiel-CGI

Im unteren Bereich der Anzeige sind Hinweise enthalten, die zum einen Aufschluss über die Arbeitsweise dieser kleinen Anwendung geben, aber auch einige Fehlerhinweise enthalten, die sich auf potenzielle Fehler beziehen, die mit dieser Anwendung auftreten können. Mit einem Klick auf die Schaltfläche "Send" schicken wir die Anforderung an den Webserver auf der i5. Der Webserver verarbeitet die Eingabefelder und liest mit Hilfe eines integrierten RPGLE-Aufrufs den Inhalt des Systemwerts QCCSID aus sowie den aktuell auf der i5 installierten i5/OS-Betriebssystemstand. Das Ergebnis wird mit einer erneuten HTML-Ausgabe auf dem Bildschirm dargestellt.



CGI-Anwendungen



Seite 1

#### 🕘 HTTP server functionality for CGIs was just tested - Microsoft Internet Explorer Datei Bearbeiten Ansicht Favoriten Extras 🤇 Zurück 💌 🔎 Suchen 🛛 👷 Favoriten 🦃 Adresse 🙆 http://soepro:8014/cgidev2p/hello1.pgm HTTP server functionality for CGIs was just tested on this system Hello! Willi your first name is Winzig vour last name is this OS/400 release is V05R03M00 Warning: system value QCCSID is set to 65535. This will inhibit characters translation in your CGI's and may result in displaying wrong characters to the browsers. Change this system value (e.g. 37 for U.S. English). IPL not required, "usually" no impact to production.

## Das Ergebnis der Programmausführung

Man beachte in der Anzeige den Warnhinweis bezüglich des verwendeten Zeichensatzes. 65535 bedeutet, dass keine Umsetzung stattfindet. Das ist leider allzu oft der Fall und auch Grund für fehlerhafte Umsetzungen etwa von Umlauten. Aber das ist ein leidiges Thema, das in diesem Zusammenhang nicht behandelt werden soll.

So einfach und trivial dieses Beispiel auch erscheinen mag, es beinhaltet bereits einige der wesentlichen Elemente, die wir in Bezug auf den Zugriff auf i5-Daten benötigen, nämlich die Eingabe von Werten, die wir z. B. als Eingabevariablen verwenden können, den Aufruf eines RPG-Programms. Hier ist der Aufruf nicht auf RPG Programme beschränkt. Alternativ kann zu einem RPG Programm auch ein CL- oder Cobol Programm eingesetzt werden. Grundsätzlich lassen sich alle auf der i5 ausführbaren Programme in ein solches CGI-Programm einbinden und mit ihm ausführen. Nachfolgend finden Sie den RPG-Code, der in diesem Programm ausgeführt wird, um die Variablen zu verarbeiten und auch den Systemwert auszulesen. Diesen RPG-Code finden wir ebenso wie den erforderlichen HTML-Code direkt in der Beispielwebseite, er ist dort als Link einsehbar.





CGI-Anwendungen

## 5.5.2 Seite 12

## Das RPG-Beispielprogramm

Schauen wir uns noch ein weiteres Beispiel an, das Inhalte aus einer i5-Datei ausliest und sie in einem subfileähnlichen Format ausgibt. Auch hierbei steht uns wieder ein Eingabefeld zur Verfügung, das als Variable in den eingesetzten HTML- und RPG-Programmen verwendet wird.

Klicken Sie zum Starten dieser Beispielanwendung auf "GO" neben "Our Delivery Times, wie es die folgende Abbildung zeigt.



Das Beispiel "Our delivery times"



	-		
Web-	Prod	rammi	eruna

Nach dem Start der Anwendung werden wir zunächst aufgefordert, einen Namen einzugeben. Dieser Name wird als Variable für die aufzurufenden Programme verwendet. Nach dem Klick auf die Schaltfläche "Enter" wird ein RPGLE-Programm aufgerufen, das aus einer i5-Datei die darin befindlichen Datensätze ausliest und sie als Liste in einem subfileähnlichen Format ausgibt.



Eingabe einer Variablen





CGI-Anwendungen

5.5.2 Seite 14 Das Ergebnis dieser Programmausführung sehen wir anschließend in unserem Browser neben den einzelnen Datensätzen, die aus der i5-Datei CTRDVY stammen.

🕘 O u	r delivery times - Microsof	t Internet Explore	r					-	
Datei	Bearbeiten Ansicht Favori	ten Extras ?							<b></b>
0	Zurück 🝷 🕥 🕤 🔀 🛃	🏠 🔎 Suchen	쑷 Favoriten	$\boldsymbol{\varTheta}$	<b>@</b> • 🎍		📄 除	🐺 🚳	
Adress	e 🝯 http://soepro:8014/cgidev	2p/ctrdvy.pgm?reques	t=go&yrname=Wi	nzig			- 💌 🔁 '	Vechseln zu	Links »
			Displa	y the H	ITML sou	rce of th	is page		^
_			<u>Display t</u>	he RP	<u>G source</u>	of this p	program		
Οι	ur delivery time	es							
Hi M	linzial								
Here	come our delivery times:								
		Deliverytime							
	Country name	(days)							
1	Afghanistan	25							
2	Albania	7							
3	Algeria	12							
4	Andorra	7							
5	Angola	25							≡
6	Antigua_and_Barbuda	17							
7	Argentina	12							
8	Armenia	22							
9	Australia	8							
10	Austria	3							
F	osition to	Send							
	_	Dana daum							
	L	Page down							
							ſ	0.020 se	
æ						<b>6</b> 2			
e							okales Intrane	et	

Das Ergebnis zeigt die i5-Daten

Achten Sie neben den Informationen, die aus der i5-Datei stammen, auch auf den unteren rechten Bereich der Anzeige, der uns Auskunft über die Ausführungsgeschwindigkeit der HTML-Abfrage gibt.



Seite 15

Diese Anzeige wird durch das folgende Codefragment innerhalb der HTML-Codequelle erzeugt:

136.00 137.00 **/\$runtime \*\*\*\* RESPONSE TIME** 138.00 139.00 140.00 <font face="Arial" size="-1"> 141.00 **/%RUNTIME%/** sec.s 142.00 
143.00

Runtime-Codefragment

Wir können in dieser Ergebnisanzeige nun wie gewohnt blättern, indem wir auf die Schaltfläche "Page Down" klicken. Alternativ dazu lässt sich mit einem weiteren Eingabefeld die Startposition innerhalb der Auflistung festlegen. Wenn wir z. B. mit dem Eintrag "Germany" in der Liste starten wollen, dann geben wir dazu im Feld "Position to" den Wert "Germany" ein und klicken auf die Schaltfläche "Send".

	Country name	Delivery time (days)
11	Azerbaijan	22
12	Bahamas	10
13	Bahrain	9
14	Bangladesh	14
15	Barbados	12
16	Belarus	10
17	Belgium	5
18	Belize	25
19	Benin	27
20	Bhutan	32
05	i <mark>tion to</mark> Germany	Send

Angabe der Startposition





CGI-Anwendungen

5.5.2 Seite 16 Mit dieser Eingabevariablen teilen wir dem Programm den Startpunkt innerhalb der Auflistung mit. Auch in diesem Fall können wir mit einem Klick auf "Display HTML Source" bzw. "Display RPG Source" die Quellen direkt einsehen, die mit dieser kleinen Beispielanwendung zum Einsatz kommen.

🖉 O u Datei	<mark>r delivery ti</mark> mes - Mio Bearbeiten Ansicht	c <mark>rosoft Internet Explor</mark> Favoriten Extras ?	er							
<b>(</b> ]	😋 Zurück 🔹 🕥 - 📓 😭 🔎 Suchen 👷 Favoriten 🤣 🍰 🍇 🔯 🗉 🗾 🐘									
Adress	Adresse 🕘 http://soepro:8014/cgidev2p/ctrdvy.pgm									
OL Hi W Here	Ir delivery t	imes es: Delivery time	Display Display th	<u>/ the HTI</u> ne RPG	ML sour source	<u>ce of</u> of this	<u>this pag</u> s prograr	e n		
	Country name	(days)								
63	Germany	10								
64	Ghana	10								
65	Greece	10								

### Einsehen der Programmcodes

Natürlich sind das zwei abstrakte Beispiele. Aber sie können bereits grundsätzlich als Vorlage für erste Anwendungen eingesetzt werden, mit denen Sie i5-Daten in Webanwendungen integrieren können. Diese Webanwendungen lassen sich dann auf dem Webserver der i5 problemlos ausführen.

Die entsprechenden Sourcen zu diesen Programmen befinden sich in der Bibliothek CGIDEV2. In ihr ist u.a. eine Quelldatei QRPGLESRC enthalten, in der wir auch die RPGLE-Programme der zuvor gezeigten Beispielanwendungen finden. Die HTML-Sourcen befinden sich ebenfalls in dieser Bibliothek und können durch Kopieren für eigene Anwendungen verwendet werden.

Ohne zu sehr ins Detail zu gehen, wollen wir uns den HTML-Codebereich doch kurz einmal anschauen.

Bei den Anweisungen "/\$top", "\$tabrow" und "\$bottom" handelt es sich nicht um HTML-Tags, sondern um spezielle Anweisungen, die das CGIDEV2-Tool benötigt, um die entsprechenden Sektionen in HTML zuzuordnen. Mit "/\$pageup" und "\$pagedowN" lassen sich die Blätteraktionen innerhalb der Anzeige realisieren.





Seite 17

Learn fro	m sources
1.00	/Stop **** START HTML
2.00	Content-type: text/html
3.00	Expires: Fri Jan 01 00:00:00 1999
4.00	
5.00	<html></html>
6.00	<head></head>
7.00	<title>Our delivery times</title>
8.00	<meta content="text/html; charset=utf-8" http-equiv="Content-Type"/>
9.00	<meta name="DOCUMENTCOUNTRYCODE" value="it"/>
10.00	<meta name="DOCUMENTLANGUAGECODE" value="it"/>
11.00	
12.00	
13.00	<body alink="blue" bgcolor="white" link="blue" text="black" vlink="blue"></body>
14.00	
15.00	/Şstart
16.00	
17.00	<font face="Helvetica,Helv,Arial,Sans-Serif"></font>
18.00	<a <="" href="/cgidev2p/dspsrc.pgm?cgiinp01=cgidev2&amp;cgiinp02=demohtml&amp;cgiinp03=ctrdvy" td=""></a>
19.00	target="_blank">Display
20.00	the HTML source of this page
21.00	
22.00	
23.00	font face="Helvetica, Helv, Arial, Sans-Serif">
24.00	
25.00	<pre><font color="blue"></font></pre>
26.00	<h2>Ask for our delivery times</h2>
27.00	<pre><iont size="-1"> </iont></pre>
20.00	<pre><iorm action-"="" ccrdvy.pgm"="" cgidevzp="" method-get=""> </iorm></pre>
29.00	<input name="request" type="nidden" value="go"/>
30.00	Later your name
31.00	<pre><int ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;<="" cicc="16" face-courier="" size-system;="" td=""></int></pre>
33 00	variand bype-cext name yrhame Size-'13"
34 00	and mease
35 00	and press singuit type=submit value="inbsn:Enterinbsn:">
36.00	
37.00	
38.00	

Codeausschnitt aus dem HTML-Bereich

Die eigentlichen Datenzugriffe erfolgen in einem RPGLE-Programm namens CTRDVY, das z.B. mit dem folgenden HTML-Codebereich aufgerufen wird. Im Bereich "/\$go" befinden sich die Tag-Anweisungen, mit denen wir auch die Codebeispiele einsehen können.

```
40.00
        /Sgo
41.00
        42.00 dlign=right><font face="flelvetica,Helv,Arial,Sans-Serif">43.00 <a href="/cgidev2p/dspsrc.pgm?cgiinp01=cgidev2&cgiinp02=demohtml&cgiinp03=ctrdvy"</td>
44.00
                  target="_blank">Display
45.00
                the HTML source of this page </font>
46.00
47.00
                 48.00 <font face="Helvetica,Helv,Arial,Sans-Serif">
49.00 <a href="/cgidev2p/dspsrc.pgm?cgiinp01=cgidev2&cgiinp02=qrpglesrc&cgiinp03=ctrdvy"</pre>
50.00
                 target="_blank">Display
51.00
                 the RPG source of this program
52.00
                </font>
53.00
                 54.00 face="Helvetica, Helv, Arial, Sans-Serif" color="blue">55.00 <h2>Our delivery times</h2>
56.00
                </font>
57.00 <font face="Helvetica,Helv,Arial,Sans-Serif" size="-1">
58.00 Hi <font color="red"><b>/%YOURNAME%/</b></font>!<br>
59.00
                Here come our delivery times:<br>
60.00
                61.00
```





5.5.2 Seite 18 Mit den folgenden Anweisungen wird das RPGLE-Programm CRTDVY aufgerufen, das die Daten aus der i5-Datei ermittelt und sie via Variablen zurück an die HTML-Seite überträgt.

/Ştabend **** END TABLE
<pre><font face="Helvetica.Helv.Arial.Sans-Serif"></font></pre>
<form action="/cgidev2p/ctrdvy.pgm" method="post" name="form1"></form>
<input name="request" type="hidden" value="go"/>
<input name="yrname" type="hidden" value="/%YOURNAME%/"/>
<input name="linesinpage" type="hidden" value="/%LINESINPAG%/"/>
Position to
<font face="courier"></font>
<input maxlength="15" name="pos" size="15" to="" type="text"/>
<input type="submit" value="%nbsp;Send "/>
<font face="Helvetica, Helv, Arial, Sans-Serif"></font>
/Spageup
<form action="/cgidev2p/ctrdvy.pgm" method="post" name="form2"></form>
<pre><input name="request" type="hidden" value="go"/></pre>
<input name="yrname" type="hidden" value="/%YOURNAME%/"/>
<input name="lastcountry" type="hidden" value="/%LASTCOUNTRY%/"/>
<pre><input name="lastseqno" type="hidden" value="/%LASTSEQN0%/"/></pre>
<pre><input name="linesinpage" type="hidden" value="/%LINESINPAG%/"/></pre>
<input name="pageupdown" type="hidden" value="pageup"/>
<pre><input type="submit" value=" Page up "/></pre>

### Der Aufruf des RPGLE-Programms

Das waren die wesentlichen Komponenten, die im Zusammenhang mit der Einbindung von i5-Daten in diesem einfachen HTML-Programm von Bedeutung sind. Sicher gibt es noch eine ganze Reihe anderer wichtiger Komponenten in diesem Programmablauf. Aber wir können und wollen an dieser Stelle den Code nicht zu sehr vertiefen. Die beiden Beispiele sollten nur ein wenig verdeutlichen, wie einfach es sein kann, in eine Webseite i5-Daten zu integrieren.

Zwangsläufig kommt in diesem Zusammenhang natürlich auch die Frage auf "Wofür brauche ich dann noch einen Anwendungsserver?". Sie lässt sich nicht allgemein beantworten. Eines ist aber als Regel zu beachten: Bei der Ausführung von CGI-Programmen auf einem Webserver lassen sich einfache(re) Anwendungen realisieren, die ohne Verfügbarkeitsaspekte und zusätzliche Sicherheitsmechanismen eingesetzt werden können. Sobald eine Anwendung aber an Komplexität gewinnt und die mit der Anwendung zu verarbeitenden Informationen auch gegen einen unberechtigten Zugriff geschützt werden sollen, wird sich der Einsatz eines zusätzlichen Anwendungsservers auf Dauer nicht vermeiden lassen. Denn wie der Name bereits sagt, ist ein Anwendungsserver dafür ausgelegt, Anwendungen auszuführen und alle damit in Verbindung stehenden Aufgaben und Funktionen bereitzustellen.



# **C** Kapitel

# 5.5.3 Seite 1

# 5.5.3 Integrierte Web Services

Erweiterungen von RPG-Anwendungen sind Bestandteile einiger Release-Versionen des Betriebssystems IBM i. Der Einsatz von ILE-Technologien erlaubt es den Betreibern der Maschine, Anwendungen zu erstellen, die zum Teil weit über das Leistungsspektrum der klassischen RPG-Funktionen hinausgehen.

"Unterstützung grafischer Oberflächen" und "Internetpräsenz" sind die Anforderungen, die von Unternehmen, die ihre Anwendungen aufwerten wollen, mit am häufigsten genannt werden. Oftmals stellt sich den IT-Entscheidern in diesem Zusammenhang die Frage, wie man RPG-Anwendungen – ausgestattet mit klassischen 5250-Green-Screen-Bildschirmen – auf eine modernere Oberfläche umstellen kann, um sie einer Benutzerschicht zugänglich zu machen, die keine Emulation für den Zugriff auf die "AS/400" nutzen kann.

Die Anzahl der zur Verfügung stehenden Technologien ist groß – und einige haben Sie bereits in den vorhergehenden Kapiteln kennengelernt. An dieser Stelle möchte ich Ihnen einen Einblick in Web Services geben, die als Standard in der Internetentwicklung weit verbreitet sind. Sie werden schnell feststellen, dass Web Services keine besonderen Schwierigkeiten darstellen. IBM bietet den System i-Betreibern eine vordefinierte Administrationsfunktion, mit der man auch ohne spezielle Web Service-Kenntnisse die IBM i-Welt (in unserem Beispiel bezogen auf RPG) und das Web-Universum miteinander verbinden kann.

Nachfolgend finden Sie Informationen, um ein RPG ILE-Programm als Web Service auf dem IBM i-Server zur Verfügung zu stellen.

### Anmerkung:

Derzeit (Stand Herbst 2011) unterstützt die IBM die Einbindung von ILE RPG-, ILE Cobol- und Service-Programmen als Web Services zusammen mit dem integrierten Web Service Server. Andere Programme – wie beispielsweise CL-Programme – werden derzeit nicht unterstützt!

Die integrierten Web Services sind bei IBM im Administrationsbereich für Web- und Anwendungsserver enthalten. Dort findet sich neben dem Verwaltungsbereich für HTTP Web Server und Anwendungsserver auch der Verwaltungsbereich für den Web Service Server.



5.5.3 Seite 2

#### Anmerkung:

Für die Ausführung von Web Services auf dem IBM i-Server sind verschiedene Voraussetzungen zu erfüllen. In diesem Beitrag finden Sie die Anleitungsdefinition für den Web Services Server. Unter anderem werden die Aufrufangaben für die Einbindung von RPG-Programmen und die dazugehörigen Parameter festgelegt. Die Ausführung der Web Services bedarf zudem eines Web Servers. Dieser ist ebenfalls auf dem IBM i-Server verfügbar. Eine entsprechende HTTP Server-Instanz wird mit der Definition eines Web Service Servers automatisch angelegt.

Die Verwaltung wird mit Hilfe eines Konfigurationsassistenten erleichtert. Diesen können wir im Browser über die Eingabe der folgenden URL starten: http://hostname:2001/HTTPAdmin. Ersetzen Sie den Platzhalter "hostname" durch den Hostnamen des IBM i-Servers!

Wenn Sie zur Eingabe eines Benutzers aufgefordert werden, dann geben Sie ein IBM i-Benutzerprofil ein, das über ausreichende Berechtigungen verfügt. Für die Verwaltung der Web Service-Dienste sind dies mindestens: \*ALLOBJ und \*IOSYSCFG.

Nach der Anmeldung gelangen wir in den Übersichtsbereich der Web- und Anwendungsserver des Systems.

HTTP-Server-Administration auf	f ASZE	BIS - Window	s Internet Explorer				
	TTPAdmi	n				🗸 😽 🗙 📴 Bir	۰ ۱۹
Datei Bearbeiten Ansicht Eavoriten	Extra	as ?					
🔶 Favoriten 🛛 🚕 🖉 Vorgeschlagene S	Sites •	Web Slice-K	atalon 💌				
A LITTO Service Administration auf ASZEPT	re					<b>*</b> • • •	🚔 🔹 Seite x Sicherheit x Evtras x 🙉 x 🎽
C ni IP-Server-Administration aur ASZEDI	6						Ball Delice - Dighermerc - Exglas -
IBM Web Administration for i							WebSphere. IBM
Konfiguration Verwaltung Erwo	ertert	Zugehorige Li	nks				
Alle Server HTTF-Server   Appli	au	Server					
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> </ul>	Alle	Server ver	rwalten 🔮				<u></u>
D HTTP-Server erstellen	Alle	HTTP-Server	Alle Anwendungsserv	er			
3 WebSphere Portal erstellen							
	Stand	1 vom 13.08.20	11 06:45:34.	<b>C</b> + +	AL D.(	7 1 4 4 1	
		Server 🔺	Version	Status	Adresse:Port	Zugeoraneter Anwendungsserver	Besch
	0		Apache/2.2.11 (i5)	Gestartet	*:00	Keine	ADMIN-Server
	0		Apache/2.2.11 (i5)	Gestoppt	.00	Keine	von ibm generenen beispier-HTTP-Server (p
	0		Apache/2.2.11 (i5)	Costoppt	*:0004		=
		WEBSERVER	Apache/2 2 11 (i5)	Gestoppt	*:80	Keine	
	0	WOI WI77	Apache/2 2 11 (i5)	Gestoppt	*11331	WOI WI77	
	ŏ	WSERVICE	Apache/2.2.11 (i5)	Gestoppt	*:10010	WSERVICE	Mit dem Assistenten für die Erstellung eines
	0	WSERVICE1	Apache/2.2.11 (I5)	Sestoppt	*:10021	WSERVICE1	Mit dem Assistenten für die Erstellung eines
	0	WSERVICE2	Apache/2.2.11 (i5)	Gestoppt	*:10032	WSERVICE2	Mit dem Assistenten für die Erstellung eines
	Serve	er-Startparame	ter:		6		
	<						
	Aktu	alisieren	Starten Stoppen	Neustart			
	Deta	ails verwalten	Löschen Um	nbenennen			
Fertig						🧐 Lokales Intrar	net 🧌 🔹 🔍 100% 🔹 🛒

### HTTP Admin starten

Wenn Sie in Ihrem Netzwerk SSL konfiguriert haben, dann verwenden Sie an Stelle des Ports "2001" den Port "2010" für eine SSL-Verbindung und geben an Stelle von "http" "https" an: https://hostname:2010/HTTPAdmin.



Kapitel

In der Einstiegsmaske des HTTP-Administrationsbereichs befindet sich u. a. auch der Eintrag "Web Service Server erstellen". Ein Beispiel dafür finden Sie in der folgenden Abbildung.

CHTTP-Server-Administration auf ASZEBIS - Windows Internet Explorer									
🚱 🕤 💌 🖻 http://aszebis:2001/HTTPAdmin									
Datei Bearbeiten Ansicht Eavoriten Extras ?									
🖕 Favoriten 🙀 🍘 Vorgeschlagene Sites 👻 😰 Web Slice-Katalog 👻									
HTTP-Server-Administration auf ASZEBIS									
IBM Web Administration for i         Konfiguration       Verwaltung         Erweitert       Zugehörige Links         Alle Server       HTTP-Server         Allegemeine Tasks und Assistenter       Alle Server verwalten         Web Services Server erstellen       Image: Server verwalten         Web Servicer stellen       Alle HTTP-Server         Alle Anwendungsserver       Alle Anwendungsserver									
Stand vom 13.08.2011 06:45:34.									
Server 🔺 Version Status Adresse:Port Zugeordneter Anwendur									
	0	ADMIN	Apache/2.2.11 (i5)	Gestartet	*:2001	Keine			
	0	APACHEDFT	Apache/2.2.11 (i5)	9 Gestoppt	*:80	Keine			

"Web Service Server erstellen"

Über diese Auswahl gelangt man in den Verwaltungsbereich der Web Service Server.



5.5.3 Seite 4

#### 5.5.3.1 Vergabe eines Server-Namens für Web Services

Auf dem IBM i-Server können mehrere Server-Instanzen angelegt werden. Diese unterscheiden sich jeweils durch einen eindeutigen Namen, der bei der Einrichtung des Servers vergeben wird. Diese Einstellungen werden – wie die übrigen Konfigurationseinstellungen auch – im IFS gespeichert. Guter Stil wäre eine optionale Beschreibung, die möglichst Auskunft über den Einsatzbereich des neuen Servers geben soll.

Für unser Beispiel wird als Name für die Web Service Server-Instanz DEMOWS01 verwendet und auch eine entsprechende Beschreibung angegeben.

Or interference in the interference	HTTP-Server-Administration aut	ASZEBIS - Window	rs Internet Explorer 📃 🗖 🔀									
Date       Berdeten       Grekkt       Evrort       Evrort       Web Sterk stadig         If the Server Administration for f       Web Sterk stadig       If the Server Administration for f       If the Server Administratis administration for f       If the S	💽 🗢 🙋 http://aszebis:2001/H1	TPAdmin										
Fevrate Fe	Datei Bearbeiten Ansicht Eavoriten	Extras ?										
Image: Control of the Control of th	🖕 Favoriten 🛛 🚖 🏉 Vorgeschlagene :	👷 Favoriten   🎪 💋 Vorgeschlagens Stes * 🔊 Web Stes-Katalog *										
IBM Web Administration for 1       Image: Termine 1 (20gebringe Links)       Image: Termine 1 (20gebringe Links)         Alle Server       HTP-Server   Application Server       Application Server         * Allegemeine Tasks und Assistents       Web Services Server erstellen         * Web Service stellen       Specify Web services server name - Schrift 1 von 9         * Anvendingsserver erstellen       Specify Web Services Server wizard. A Web Services server provides a convenient way to externalize existing programs running on IBM I, such as RPC and COBOL programs, sub Web services server provides a convenient way to externalize existing programs running on IBM I, such as Service Server as SUB-Services server provides a convenient way to externalize existing programs running on IBM I, such as Service Server as SUB-Services server provides a convenient way to externalize existing programs running on IBM I, such as Service Server as Sub Services server provides a convenient way to externalize existing programs running on IBM I, such as Service Server as Sub Services server provides a convenient way to externalize existing programs running on IBM I, such as Service Server as Sub Services server provides a convenient way to externalize existing programs and programming languages such as C, C, et al. As and NET This Watad recease exerpting needed to nu Web Services. Web services server of Server asset as Sub Services Server of Server asset as Sub Services Server of Server asset asset as Sub Service Server of Server asset asset asset asset as Sub Service Server of Server asset asse	HTTP-Server-Administration auf ASZEBI	s	🦄 🔹 🗟 👘 🔹 Sejte + Sigherheit + Extras + 🔞 + 🎽									
Alle Server       HTTP-Server   Application Server         * Allgemeine Taskis und Assistenten B HTTP-Server erstellen B Web Services Server erstellen B Web Services Server erstellen B Web Services server name - Schrift 1 von 9         * Web Services and COBOL programs, as Web services server provides a convenient way to externalize existing programs running on IBM I, such as RPG and COBOL programs, as Web services. Web services server provides a convenient way to externalize existing programs running on IBM I, such as RPG and COBOL programs, as Web services. Web services uch as SOAP: The clients can be implemented using a variety of platforms and programming languages such as SO. C+-, Java and NET. This wizard creates everything needed to run Web services.         * Uber Service Server of the Create Web Service Server of this server of service has service from the internet of infrart os service languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages such as SOAP. The clients can be implemented using a variety of platforms and programming languages use has client the client can be implemented using a variety of platforms and programming languages use has client the client can be implemented using a variety of platforms and programming languages use has c	IBM Web Administration for i Konfiguration Verwaltung Erw	eitert   Zugehörige L	inks WetSphere IBM									
<ul> <li>Allgemeine Tasks und Assistenten Web Services Server erstellen HTTS-erver erstellen WebSphere Portal erstellen</li> <li>WebSphere Portal erstellen WebSphere Portal erstellen</li> <li>WebSphere Portal erstellen</li> <li>Server aus C. C+-, Java and NET This wizard creates everything needed to run Web services.</li> <li>Weitere Informationen erhalten Sie auf folgenden Seiten: http://www-03.bm.com/systems/l/software/ws/</li> <li>Specify a unique name for this server ?</li> <li>Server description: Web Service Server 01 für Test</li> <li>Zurück Weiter</li> <li>Abbrechen</li> </ul>	Alle Server HTTP-Server   Appli	cation Server										
	Allgemeine Tasks und Assistenten     Web Services Server estellen     HTTP-Server estellen     Anwendungsserver erstellen     WebSphere Portal erstellen     WebSphere Portal erstellen	Web Services Specify Web servi Welcome to the Crr RPG and COBOLp service based indu languages such as Weltere Information Specify a unique n Server name: Server description:	Server erstellen         ces server name - Schrift 1 von 9         hate Web Services Server witzard. A Web services server provides a convenient way to externalize existing programs running on IBM I. such as organise, as Web services. Web services in then interact with these IBM i program based services from the internet or intranet via Web stry standard controls us una stopA. The clients can be implemented using a variety of platforms and programming IC, C++, Java and NET. This witzard creates everything needed to run Web services.         use rehaten Sie auf folgenden Setten: http://www-03.ibm.com/systems//softwarel/ws/         ame for this server           DemoWS01           Web Senice Server 01 für Test									
Fertig	Fertig		🔮 Lokales Intranet 🎻 🔹 😤 100% 🔹									

Neue Web Service Server-Instanz erstellen

Nach Eingabe der notwendigen Parameter wird die Konfiguration mit einem Klick auf die Schaltfläche "Weiter" fortgesetzt.



## 5.5.3.2 Angabe der Benutzer-ID

Die Ausführung des IBM i-Jobs bedarf eines Benutzerprofils. Alle Jobs laufen auf dem IBM i-Server unter einem spezifischen Benutzer. Damit werden u. a. Berechtigungen und Ressourcenzuordnungen gesteuert. Für die Ausführung der Web Services muss eine Benutzervorgabe erfolgen. Damit werden alle Web Service-Jobs unter dem angegebenen Benutzer ausgeführt. IBM liefert für die Zuordnung der Web Services ein Pseudo-Benutzerprofil QWSERVICE aus, das verwendet werden kann. Oder Sie geben ein individuelles Benutzerprofil an. Wichtig ist hierbei jedoch, dass das individuelle Profil aktiv ist und man sich am System anmelden kann!

Anmerkung: Die Standard-Benutzer-ID für den Server ist QWSERVICE C Eine vorhandene Benutzer-ID angeben C Eine neue Benutzer-ID angeben Zurück Weiter Abbrechen 📢 Lokales Intranei

Benutzer-ID

Nach Eingabe des Benutzerprofils setzen wir die Konfiguration mit einem Klick auf "Weiter" fort.





Seite 5

5.5.3 Seite 6

## 5.5.3.3 Deployen eines RPG-Programms als Web Service

Die Bereitstellung eines ILE- oder Service-Programms (unterstützt werden RPG und Cobol) als Web Service kann gleich bei der Neuanlage eines Servers oder aber auch später erfolgen. Für dieses erste Beispiel übernehmen wir die Standardeinstellung, wie sie in der folgenden Abbildung zu sehen ist. In Abhängigkeit von der hier getroffenen Auswahl variieren die Folgeanzeigen. Nach Übernahme der Standardwerte wird die Konfiguration mit "Weiter" fortgesetzt.



*IBM i-Programm als Web Service auslagern* 



# 5.5.3.4 **Details zum IBM i-Programm**

Für die Verwendung von IBM i-Programmen eignen sich ILE RPG- oder Cobol-Programme. Programme, die einen Web Service einbinden, müssen entsprechend vorbereitet und angepasst werden. Diese notwendigen Anpassungen beschränken sich allerdings nur auf einige wenige Codezeilen, wie Sie im Folgenden sehen werden.

Die Einbindung eines ILE-Programms als Web Service erfolgt auf Basis von PCML-Dokumenten. PCML steht für "Program Call Markup Language" und ist eine in XML erzeugte Beschreibung für die Ein- und Ausgabeparameter des Services, die dieser an das aufrufende Programm übergeben muss. Die für die Verwendung von PCML notwendigen Informationen werden vom Konfigurationsassistenten abgerufen und entsprechend gespeichert. Bei der Neuanlage eines Services müssen die Parameter einmalig angegeben und bei Bedarf später auch angepasst werden.

Für die Erstellung der PCML-Informationen können unterschiedliche Werkzeuge genutzt werden:

- 1. Einsatz eines Texteditors.
- Verwendung von RDp for IBM i. 2.
- Kompilieren des ILE-Moduls. 3.

Damit werden die PCML-Informationen direkt in das Modulobjekt eingebunden und können von dort aus verwendet werden.

Die PCML-Informationen werden bei der Kompilierung über den Parameter "Program Interface Information" (PGMINFO) gesteuert. Dieser Parameter steht für die Befehle CRTRPGMOD und CRTCBLMOD zur Verfügung. Im folgenden Beispiel habe ich das Kompilieren mit RDp durchgeführt. Natürlich steht der PCML-Parameter auch bei der Wandlung mit PDM-Funktionen zur Verfügung. Da wir jedoch Web Services entwickeln wollen, bietet sich hier der konsequente Einsatz eines Werkzeugs an, mit dem alle Komponenten der modernen Anwendungsentwicklung genutzt werden können!











# Web-Programmierung

Integrierte Web Services

5.5.3	RPG-Modul erstellen (CRTRPGMO	D)	X
C .: (	ivioaui ersetzen:	> '1ED C31' <	^
Seite 8	Berechtigung:	*LIBCRTAUT	Name
	Numerische Daten abschneiden:	*YES 🗸	
	Numerische Daten korrigieren:	Hinzufügen	
	Ziel-Release:	*CURRENT	
	Nullwerte zulassen:	*NO 🗸	
	Bedingungsnamen definieren:		✓ Hinzufügen
	Leistungsdatenerfassung aktiv.:	*PEP 🔽	
	Profildefinitionsdaten:	*NOCOL	
	Binderverzeichnis:	*NONE	Name
	Bibliothek:	*LIBL V Hinzufügen	Name
	LIC-Optiopen:		Zeichenwert
	Erc-optionen.		
	Verzeichnis einschließen:		V Hinzufügen
	ProgSchnittstelleninfo:		
	Generieren:	*PCML 🗸	
	Position:	*MODULE	
	Datenstromdatei f. ProgSchn.:		
	Vorprozessoroptionen:	Hinzufügen	
	Ausgabeguellendatei:		Name
	Bibliotbek	*CLIRI TB	Name 👱
	<u>&lt;</u>		>
	Erweitert 🔽 Alle Parameter 📃 :	Schlüsselwörter	
	CRTRPGMOD MODULE(ZEIG/QIWSSAMPLE) S DBGVIEW(*SOURCE) REPLACE(*YES) PGMIN	RCFILE(ZEIG/QRPGLESRC) SRCMBR(QIWSSAMPLE) OPTION FO(*PCML *MODULE)	J(*EVENTF)
		OK Standardwerte wiederherstellen	Abbrechen

#### PCML-Informationen

RPG hat dieses Beispiel als Modul ausgeliefert. Aus diesem Grund verwenden wir für das Kompilieren den Befehl CRTRPGMOD. Man kann den Programmcode auch so anpassen, dass man direkt ein Programmobjekt erstellen kann. Ich habe an dieser Stelle bewusst auf das Anpassen des Codes verzichtet. Damit der Code als Programm ausgeführt werden kann, erstellen wir nun noch als Zwischenschritt ein ausführbares Programm. Dazu verwenden wir den Befehl **CRTPGM:** 


**Integrierte Web Services** 



Seite 9

90 Sitzung A - [24 x 80]	
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hillfe	
Programm erstellen (CRTPGM)	
Auswahl eingeben und Eingabetaste drücken.	
Programm	
Bibliothek zeig Name. *CURL	IB
Modul	isch*. *PGM. *ALL
Bibliothek	*CURLIB
+ für weitere Werte	,
Text 'Beschreibung' *ENTMODIXT	
Text besch erbang i i i i i i i <u>«Enthobrar</u>	
	Ende
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F10=Zusätz	l. Parameter
F12=Abbrechen F13=Verwendung der Anzeige F24=Weitere Tas	ten
MA MW	08/043
្យា <sup>ព</sup> I902 - Sitzung wurde erfolgreich gestartet.	1

Programmobjekt erstellen

Für die Einbindung eines Web Services können auch mehrere Programme auf dem IBM i-Server genutzt werden. Dazu ist es erforderlich, dass die einzelnen PCML-Anweisungen in ein einzelnes PCML-Dokument eingebunden werden, das folgendes Format aufweisen soll:

```
<program name="p1" ... > ... </program>
<program name="p2" ... > ... </program>
</program>
```

PCML-Dokumente können bei Bedarf auch manuell angepasst werden. Dabei müssen allerdings die PCML-Regeln genau eingehalten werden, denn anderenfalls wären die Ergebnisse falsch.

Sollten Sie unsicher im Umgang mit PCML sein, dann empfehle ich die Verwendung der PCML-Dateien, die als Standard von den Kompilierbefehlen erzeugt werden.

Ab Version "V6R1" werden die PCML-Informationen mit den Kompilierbefehlen erzeugt. Beispiele dazu sind:

```
CRTRPGMOD PGMINFO(*PCML *MODULE)
CRTCBLMOD PGMINFO(*PCML *MODULE)
```





**Integrierte Web Services** 

5.5.3 Seite 10 Wenn Sie die Betriebssystemversion "V5R4" verwenden (ACHTUNG! Spezielle PTFs sind hier erforderlich!), dann muss im Header-Bereich von RPG die folgende Codezeile vor dem Kompilieren eingegeben werden:

H PGMINFO(\*PCML:\*MODULE)

IBM liefert zusammen mit dem integrierten Web Service Server Beispielanwendungen aus.

Als Beispiel nutzen wir ein Programm zur Umrechnung der Temperatur von Grad in Fahrenheit. Das dazugehörige Serviceprogramm QIWSSAMPLE befindet sich in der Bibliothek QSYSDIR. Dabei handelt es sich lediglich um ein einfaches RPG-Beispiel, das für Demonstrationszwecke in einem Web Service eingebunden werden kann. Das Beispiel verfügt über einen Eingabe- und einen Ausgabeparameter. Dabei können Sie sehen, wie ein Web Service mit einem RPG-Programm erstellt werden kann.

Der Quellcode zu diesem Serviceprogramm befindet sich im Standard-Installationsverzeichnis für den Web Service Server:

/QIBM/ProdData/OS/WebServices/V1/server/samples/ConvertTemp/ CNVRTTMP.RPGLE



RPG-Beispielprogrammcode im IFS



**Integrierte Web Services** 

Kapit

Seite 11

Schauen wir uns den Beispielcode kurz an:

0	QIWSSAMPLI	E.RPGLE X				
	Zeile 2	Spalte 1 Er	setzen			
		HKeywords+++++++	++++++++	+++++++++++++++++++++++++++++++++++++++	++++++++++	++++++++++++++++++++++++++++++++++++++
H						
⊕	000101	h PGMINFO(*PCML	:*MODULE)	THREAD (*S)	ERIALIZE)	
⊢⊞	000152	d ConvertTemp	pr			
	000154	d tempIn		10	const	
l⊎	000155	d tempOut		10		
⊎	000157	p ConvertTemp	b		export	
	000159	d ConvertTemp	pi			
	000160	<mark>d</mark> tempIn		10	const	
l 🗄	000161	d tempOut		10		
	000163	d tempI	s	8P	2	
	000164	d tempO	3	8P	2	
	000165	d value	s	50A		
	000166	/free				
	000167	value = %STR(%.	ADDR (temp	In));		
	000168	tempI=%DEC(val	ue:7:2);			
	000169	tempO = (5/9)*	(tempI -	32);		
	000170	value = %CHAR(	tempO);			
	000171	tempOut = valu	=;			
	000172	%STR(%ADDR(tem)	pOut):10)	=tempOut;		
	000173	*inlr = *on;				
	000174	/end-free				
	000175	p ConvertTemp	e			

**RPG-Beispielcode** 

Das Beispiel ist – wie die meisten neueren IBM-Codebeispiele – in Free Format RPG geschrieben.

In der H-Bestimmung finden wir die PCML-Informationen. Diese Informationen sind für die Verwendung des RPG-Programms als Web Service notwendig!

In den Prototypangaben finden wir die beiden Parameter

- tempIn
- tempOut

Über die Rechenbestimmungen wird eine Berechnung ausgeführt, die die Temperaturen Grad Celsius und Fahrenheit berechnen. Wie Sie sehen, handelt es sich um ein sehr einfaches RPG-Beispielprogramm. Da es über die notwendigen PCML-Informationen verfügt, kann es nun als Web Service eingebunden werden. Dazu geben wir den Namen und die dazugehörige Bibliothek in die Eingabemaske des Konfigurationsassistenten ein. ACHTUNG! Der Konfigurationsassistent prüft, ob das angegebene Objekt tatsächlich in der Bibliothek existiert. Bestätigen Sie die Eingabe mit "Weiter". Kapitel 2

## Web-Programmierung

**Integrierte Web Services** 

5.5.3 Seite 12

Image: Server and the Build and Assistential of the Service angeben.         Image: Server and the Build and the angeben.	C HTTP-Server-Administration au	f ASZEBIS - Windows Internet Explorer			
Detail gestekten grokht gevreten Egtes 2		TTPAdmin		💌 🐓 🗙 🔽 Bing	P -
Forvater	Datei Bearbeiten Ansicht Eavoriten	Extras ?			
Image: Interpretendent of the service of	🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene	Sites 🔹 🔊 Web Slice-Katalog 👻			
IBM Web Administration for I       Image: Construction of the Service of Service	HTTP-Server-Administration auf ASZEB	IS		🦄 🔹 🗟 🕆 🖃 🊔 👻 Sejte 🕶 S	Sigherheit • Extras • 🔞 • 🂙
<ul> <li>Allgemeine Tasks und Assistente <ul> <li>Web Services Server erstellen</li> </ul> </li> <li>Anwendungsserver erstellen</li> <li>Anwendungsserver erstellen</li> <li>WebSphere Portal erstellen</li> <li></li></ul>	IBM Web Administration for i Konfiguration Verwaltung Erw Alle Server HTTP-Server   Appl	reitert   Zugehörige Links			WebSphere.
Lakalas Tatvanat 🔿 – 😤 1009/ 🖛	Allgemeine Tasks und Assistenten     Web Sprices Server erstellen     HTTB-Server erstellen     Anwendungsserver erstellen     WebSphere Portal erstellen	Web Services Server erstellen         Neuen Service implementieren: Speicherps         Das auszulagende 803-Programmobjekt ruwerden nur Programmobjekt für den We         O Den Namen der IBM i-Bibliothek und der         Die Speicherposition des Programmobjekt für den We         O Den Namen der IBM i-Bibliothek und der         Die Speicherposition des Programmobjekt für den We         Die Leicherposition des Programmobjekt für den We         Die Den Namen der IBM i-Bibliothek und der         Die Speicherposition des Programmobjekt für den Name des Programmobjekt für den Name des Programmobjekt für den Namen des Programmobjekt für den Namen des Programmobjekt für den Namen der Stepren Namen der IBM i-Bibliothek und der II         LiLE-Objektrip:       • sRvPGM @• Programmobjekt für den Namen des Programobjekt für den Namen des Programmobjekt für	bsition von IBM i-Programmobjekt angeben - muss ein auf dem System vorhandenes ILE- od e mit der Programmiersprache COBOL oder Ri ab-Service angeben. si ILE-Programmobjekts angeben (Empfohlen) ekts können Sie angeben, indem Sie sowohl d ekts selbst. Dies ist der schneliste und auch de SM BM i-Programmobjekt durchsuchen	Schrift 4 von 9 er Senkceprogrammobjekt (*PGM bzw. *SRV *G erstellt wurden. en Namen der Bibliothek eingeben, die das er empfohlene Weg zu Lokalisierung des Pro	PGM) sein. Derzeit Programmobjekt enthält, igrammobjekts.

RPG als Web Service angeben

Um ein solches ILE-Programm mit einem Web Service einsetzen zu können, müssen wir Namen und Speicherort angeben. Basierend darauf wird dann der Web Service erstellt. Für die Angabe eines Programmobjekts kann wahlweise dessen Name und die Bibliothek angegeben werden, oder das IFS kann nach dem gewünschten Objekt durchsucht werden. Letztere Option könnte jedoch einige Zeit in Anspruch nehmen und sollte deshalb nur im Notfall verwendet werden!

Mit einem Klick auf "Weiter" gelangen wir in die nächste Anzeige des Konfigurationsassistenten.



#### 5.5.3.5 Namensvergabe für den Web Service

Dem Web Service muss ein eindeutiger Name zugeordnet werden. Das wird in der nächsten Anzeige gemacht. Als Standardeinstellung verwendet der Assistent den Programmnamen, der als Web Service eingebunden werden soll. Demzufolge steht als Name in unserem Beispiel QIWSSAMPLE in der Anzeige. Der Service-Name und auch die zugehörige Bezeichnung können natürlich überschrieben werden.

CHTTP-Server-Administration aut	ASZEBIS - Windows Internet Explorer		
💽 🗢 🙋 http://aszebis:2001/HT	TPAdmin	💌 🗲 🔀 Bing	<b>P</b> -
<u>D</u> atei <u>B</u> earbeiten <u>A</u> nsicht <u>E</u> avoriten	Extras ?		
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene S	iites 🔹 🙋 Web Slice-Katalog 🔹		
HTTP-Server-Administration auf ASZEBI	5	🛅 🔹 🔝 🕤 🖃 🖶 👻 Sejte - Sigherheit - Extra	15 • 🕜 • 👋
IBM Web Administration for i Konfiguration Verwaltung Erw	eitert   Zugehörige Links	WebSphere.	IBM
Alle Server HTTP-Server   Appli	cation Server		
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> <li>HTTP-Server erstellen</li> <li>Anwendungsserver erstellen</li> <li>WebSphere Portal erstellen</li> </ul>	Web Services Server erstellen         Neuen Service implementieren:Namen für Service angeben - Schritt 5 von 9         Einen eindeutigen Namen für diesen Service angeben.         Servicename:       WSGRAD01         Servicebeschreibung:       WS Beispiel für Umrechnung Grad         Zurück       Weiter         Abbrechen		
		Sector Strate St	100% 👻 👉

Web Service-Name

Bestätigen Sie die Eingaben mit "Weiter".



apite

**Integrierte Web Services** 

5.5.3 Seite 14

#### 5.5.3.6 Auswahl der Exportprozeduren

Die Angabe der Parameter ist für Web Services in Verbindung mit dem aufrufenden Programm von besonderer Bedeutung. Der Web Service muss einen Einstiegspunkt für das anzubindende RPG-Programm haben. Dafür wird in der nächsten Anzeige lediglich ein Eintrag angezeigt. Dabei handelt es sich um die Prozedur Main, die in jedem RPG-Programm zu finden ist. Sollten Sie anstelle eines RPG-Programms ein Serviceprogramm einbinden, dann können auch mehrere Exportprozeduren angezeigt werden.

Durch Erweitern des Prozedureneintrags werden die zugehörigen Parameter zur Auswahl angezeigt und können als Ein- und Ausgabeparameter selektiert werden. Zudem wird die Art des Parameters dargestellt.

Damit die Parameter angezeigt und verwaltet werden können, wird die Anzeige mit einem Klick auf den gewünschten Service-Eintrag erweitert.



Parameterangaben



Integrierte Web Services

Folgende Selektionen sind in diesem Bereich zu tätigen:

- Im Bereich "Auswählen" legen wir fest, welche Prozeduren für den Web Service genutzt werden können. Hier können auch Prozeduren selektiert werden, die im Web Service nicht zum Einsatz kommen, allerdings ist das nicht zu empfehlen.
- Der Parameter wird festgelegt, damit das Programm in Zusammenarbeit mit dem Web Service genutzt werden kann.
- In der Spalte "Verwendung" wird angegeben, ob der betreffende Parameter als Eingabe- und/oder als Ausgabeparameter zum Einsatz kommt. Bei der Auswahl der Parameter ist darauf zu achten, dass der Web Service genau den festgelegten bedient.

Neben den Auswahlfeldern gibt es noch zwei Informationsfelder in dieser Anzeige. Sie geben Auskunft über den Datentyp des jeweiligen Parameters.

Der Eintrag "Anzahl" liefert Informationen darüber, ob es sich beim Parameter um eine Feldgruppe handelt. Wenn das der Fall ist, dann gibt es hier einen Wert, der die Anzahl der Einträge in der Feldgruppe liefert.

In unserem Beispiel ist der Parameter TEMPIN als Eingabeparameter definiert. Der Parameter TEMPOUT ist als Ausgabeparameter festgelegt. Das hat zur Folge, dass der Web Service Client nur den TEMPIN-Parameter übergeben muss. Dieser erhält dann als Ergebnis die Rückgabe des Parameters TEMPOUT.

Folglich sollten unsere Parametereinstellungen wir folgt ausschauen:

Exportprozeduren: 3							
Auswählen	Prozedurname/Parametername	Verwendung	Datentyp	Anzahl			
	▼ QIWSSAMPLE						
	▼ CONVERTTEMP						
	TEMPIN	input 💉	char				
		output 🗸 🗸	char				
Alles auswählen Alles abwählen Alles einblenden Alles ausblenden							

#### Angepasste Parameter

Die Konfiguration wird mit "Weiter" fortgesetzt.



**Integrierte Web Services** 

5.5.3 Seite 16

#### 5.5.3.7 Angabe der Benutzer-ID

Jeder Job muss mit einem vorgegebenen Benutzer auf dem IBM i-Server ausgeführt werden. Dazu ist in der Folgeanzeige eine gültige Benutzer-ID anzugeben. In der nächsten Anzeige ist es möglich, einen Standardbenutzer oder eine individuelle Benutzer-ID für die Web Services anzugeben. Natürlich gelten hier auch die allgemeinen Regeln für eine Benutzer-ID – d. h., sie muss aktiv sein und über ausreichende Berechtigungen für die gewünschten Tätigkeiten verfügen.



Service Benutzer-ID

Für unser Beispiel verwenden wir die Benutzer-ID der Servers und klicken auf die Schaltfläche "Weiter".



Web-Programmierung **Integrierte Web Services** 

# Seite 17

Analog zu den klassischen Anwendungen muss für den Aufruf der RPG-Programme eine entsprechende Umgebung definiert werden. Die Wahl der Bibliotheksliste ist hier ebenso notwendig wie im nativen Ausführungsbereich. In der nächsten Anzeige werden die Bibliotheken festgelegt, die für die Ausführung des Web Services benötigt werden. Durch Selektionsoptionen kann zudem festgelegt werden, an welche Stelle des Benutzerteils der einzugebende Bibliotheksname gestellt werden soll.

# Allgameine Tasks und Assistenten 9 Web Services Server erstellen 9 HTTP-Server erstellen 9 Anwendungsserver erstellen 9 WebSphere Portal erstellen 9 Die Funktionalität des IBM I-Programms, da

Angabe der Bibliotheksliste

5.5.3.8

Zurück Weiter Abbrechen 📢 Lokales Intranet Bibliotheksliste anpassen

Nachdem die Bibliothekseinträge hinzugefügt wurden, wird die Arbeit mit "Weiter" fortgesetzt.

Damit sind die einzelnen Schritte in der Definition des Web Services abgeschlossen. Allerdings ist er noch nicht definitiv erstellt worden. In der Folgeanzeige sehen wir eine Zusammenfassung der bisher unternommenen Arbeitsschritte.





**Integrierte Web Services** 

#### 5.5.3 Seite 18

#### 5.5.3.9 Zusammenfassung

Die Zusammenfassung der Konfigurationsangaben beinhaltet alle Teilschritte, die wir zuvor definiert haben. Prüfen Sie sie sorgfältig! Natürlich können einzelne Angaben auch später noch angepasst werden.

In der Server-Übersicht erscheint der neue Web Service Server. Außerdem findet man in der Übersichtsanzeige auch die Informationen zum automatisch miterstellten Web Server. Neben seinem Namen ist natürlich auch der Port wichtig, über den der Server erreicht werden kann.

C HTTP-Server-Administration aut	f ASZEBIS - Windows Internet Explorer		$\mathbf{X}$
💽 🗢 🙋 http://aszebis:2001/HT	TPAdmin	V 47 🗙 🔁 Bing	•
Datei Bearbeiten Ansicht Eavoriten	Extras ?		
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene S	Sites 🔹 🙋 Web Slice-Katalog 🔹		
HTTP-Server-Administration auf ASZEBI	5	🏠 👻 🔂 👘 🖃 🚋 👻 Seite 🗸 Sigherheit – Extras – 🔞 –	»
IBM Web Administration for i Konfiguration Verwaltung Erw	ellert   Zugehörige Links	() WebSphere.	
Alle Server Hitte-Server [ Appli	Calori Server		
Algemeine Tasks und Assistenten     Web Services Server erstellen     UrtTP-Server erstellen     UrtTP-Server erstellen     UrtP-Server erstellen     UrtP-Sphere Portal erstellen     UrtP-Sphere Portal erstellen	Web Services Server erstellen Neuen Service implementieren:Bibliotheksliste angeben - Schrit Die Funktionalität des IBM i-Programms, das als Web-Service ausg Geben Sie alle Bibliotheken an, die Programme enthalten, von den die Standardbullothek verwendet. Specify library list position for this Web service: O Insert libraries in front of user library portion of the library list insert libraries at the end of user library portion of the library list Bibliothekslisteneinträge: Bibliotheksname ZEIG Hinzufügen Alles entfermen	t 8 von 9 elagert werden soll, hängt möglicherweise von anderen IBM i-Programmen im System ab. en die Web-Service-Programme abhängig sind. Wenn keine Bibliotheksangabe erfolgt, wird	
	Zurück Weiter Abbrechen		
			>
Fertig		Sector Contract 🛛 🖓 🔹 🔍 100% 🔹	•

Zusammenfassung allgemein

Im Bereich "Services" finden Sie die zuvor definierten Informationen über die Web Services. Sie werden wie folgt dargestellt:

- Name
- Beschreibung
- Installationspfad
- Benutzer-ID, unter der der Service ausgeführt wird
- Eingebundenes RPG-Programm
- · Bibliotheksliste, die für den Service genutzt wird
- URL für den Web Service





Seite 19

Integrierte Web Services

CHTTP-Server-Administration auf A	ASZEBIS - Windows Inte	rnet Explorer		
💽 💽 🔻 🙋 http://aszebis:2001/HTT/	PAdmin		🗸 🐓 🗙 📴 Bing	• ٩
Datei Bearbeiten Ansicht Eavoriten	Extras ?			
🖕 Favoriten 🛛 👍 🌈 Vorgeschlagene Sit	es 👻 🖉 Web Slice-Katalog	-		
HTTP-Server-Administration auf ASZEBIS			🐴 🔹 🔝 🕤 📑 👘 🔹 Seijte 🕶	Sigherheit • Extras • 🕢 •
IBM Web Administration for i Konfiguration Verwaltung Erwei Alle Server HTTP-Server   Applici	tert   Zugehörige Links			WebSphere.
* Allgemeine Tasks und Assistenten 현 Web Services Server erstellen 한 HTTP-Server erstellen 한 Anwendungsserver erstellen 한 WebSphere Portal erstellen	Wenn Sie auf Fertig stelle Server Services Ope	n Klicken, wird alles erstellt, was zur Ausführung des Web Services rationen	Server erforderlich ist.	
1	Musterservices:			
	Musterservicename	Muster-URL für Web-Service-Definition		
	ConvertTemp	http://aszebis.zebis.de:10054/web/services/ConvertTemp?wsdl		
	Vom Benutzer implement Name: Beschreibung: Serviceinstallationspfad: Benutzer-ID für Service: Programm: Bibliotheksliste für Servic URL für Web-Service-Def Zurück Fertig steller	ierter Service: WSGR4D01 WS Beispiel für Umrechnung Grad Nww/demows01/iwi/websen/ces/services/WSGR4D01 *SERVER (OWSERVICE) /OSYS LIB/ZEI/G LIB/DIWSSAMPLE PGM ez ZEIG mittion. http://aszebis:10054/web/services/WSGR4D01?wsdl Abbrechen	Dr	uckbare Zusammenfassung
Fartic			S Lokalar Intranat	₩ 100% ×

Zusammenfassung Services

Im Tabellenbereich "Operationen" finden sich die Web Service-Operationen, die für diese Prozedur zum Einsatz kommen.

C HTTP-Server-Administration auf	ASZEBIS - Windows Internet Explo	rer		
G v E http://aszebis:2001/HT	TPAdmin		💌 🗲 🗙 🔁 Bing	P -
Datei Bearbeiten Ansicht Eavoriten	Extras ?			
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene S	Sites 🔹 🙋 Web Slice-Katalog 👻			
HTTP-Server-Administration auf ASZEBI	s		🚹 • 🖻 × 🖬 🖶 •	Seite • Sigherheit • Extras • 🕢 •
IBM Web Administration for i Konfiguration Verwaltung Erw Alle Server HTTP-Server   Appli	eitert   Zugehörige Links			WebSphere.
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li><sup>●</sup> Web Services Servier erstellen</li> <li><sup>●</sup> HTTB-Server erstellen</li> <li><sup>●</sup> Anwendungsserver erstellen</li> <li><sup>●</sup> WebSphere Portal erstellen</li> <li><sup>●</sup> WebSphere Portal erstellen</li> </ul>	Web Services Server erstit       Zusammenfassung - Schrift 9 von S       Wenn Sie auf Fertig stellen klicken, v       Server     Services       Operationen       Verfugbare Operationen für "QIWS       Operationsnamen qiwssample_XML       Zurück     Fertig stellen	llen Ird alles erstellt, was zur Ausführung des SAMPLE.PGM'' :	: Web Services Server enforderlich ist.	Druckbare Zusammenfassung
Fertig			S Lokales Intranet	🖓 • 🔍 100% • j

 $Zusammen fassung\ Operation en$ 

Wenn alle Angaben korrekt sind, dann erfolgt die Fertigstellung mit "Fertig stellen". Daraufhin werden die einzelnen Konfigurationsschritte ausgeführt.



Kapitel

#### Web-Programmierung

Integrierte Web Services

#### 5.5.3 Seite 20



Fertigstellung

Das Ergebnis sollte jetzt ein einsatzbereiter Web Service sein.

Wenn der Service fehlerfrei angelegt werden konnte, werden der Web Service Server und der zugehörige Web-Server automatisch gestartet. Der Status wird in der nächsten Anzeige angezeigt. Zusätzlich werden die beiden Server auch in der allgemeinen Serverübersicht aufgeführt; sie können von dort aus auch verwaltet werden.







Web Berkets Arskt. Evoriten Egrs 1   Foronten Foronte	🛁 🔁 🖉 http://aszebis:2001/HTTPAdmin	V G Bing
Provention  Web Science Science Full      Provention       Provide in Provide Intervide in Provide Intervide in Provide Intervide in Provide Intervide Inte	atei Bearbeiten Ansicht Eavoriten Extras ?	
ITTP-Server-Administration and ASZEBIS       Image: Server Administration for intercement of the server and the server intercement of	🗧 Favoriten 🛛 👍 🏉 Vorgeschlagene Sites 🝷 💋 We	eb Slice-Katalog 🔹
BM Web Administration for 1       Implementation	HTTP-Server-Administration auf ASZEBIS	🏠 🔹 🗟 👘 🔹 Sejte + Sigherheit + Extras + 🔞 -
Alle Sener   HTTP-Sener   Application Server Gestantel  Gestantel	BM Web Administration for i Konfiguration Verwaltung Erweitert   Zugehi	orige Links
Gestatet       Server:       DEMOVIS01-V13 (Web-Services)         Algemeine Tasks und Assistemen Web Services Server estellen WebSphere Portal erstellen       DEMOVIS01         WebSphere Portal erstellen       DEMOVIS01         WebSphere Portal erstellen       DEMOVIS01         WebSphere Portal erstellen       DEMOVIS01         WebSphere Portal erstellen       WebSphere Portal erstellen         WebSphere Portal erstellen       DEMOVIS01         WebSphere Portal erstellen       WebSphere Portal erstellen         WebSphere Portal erstellen       WebSphere Portal erstellen         Neuen Service - Assistenten DEspirachaten DESpirachaten	Alle Server   HTTP-Server Application Server	
Allgemeine Tasks und Assistenten       DEMOWSD1         Web Services Server strellen       Web Services Server strellen         Anwendungsserver erstellen       Web Service Forbal erstellen         Web-Service-Assistenten       Server: DEMOWSD1         Nueun Service implementieren       Web Senvice Server tretwalten         Service-Tigenschaften Diegver-Traing       Web Service-Service-Assistenten COBOL-Programme, als Web-Services auszulagem, Web-Service-Clients können dann unter Verwendung von Web-Service- Desvices Services anzeigen         Binglemeinter Services Verwater       Ein Web Services Server und die Services Server und die Services Services auszulagem, Web-Service-Clients können dann unter Verwendung von Web-Service- Desvices Services auszulagem, Web-Service-Services auszulagem, Web-Service-Services auszulagem, Web-Service-Services auf IBM I-Programme, als Web-Services Server und die Services Services Server und die Services Services Services Services auszulagem, Web-Service-Services auszulagem, Web-Service-Services auszulagem, Web-Service-Services auszulagem, Web-Service-Service-Services auszulagem, Web-Service-Service-Service-Services auszulagem, Web-Service-Servic	) Gestartet 💟 🔲 💋 Server: DEMOWS01	- V1.3 (Web-Senices) 💌
Web-Sandce-Assistenten       Web Senice Senier für Test         Begenschaften       Eigenschaften         Begenschaften       Eigenschaften         Begenschaften       Eigenschaften         Bereitrauf       Ein Web Senices Sener ist ein komfontable Moglichkeit, vohandene Programme, die unter IBM i laufen, wie z. B. RPG- und Web Senices Sener und die Senices für IBM i Programmobjekte konfiguriet werden konnen. Außerdem werden werden Web Berlomance Außerden werden werden werden werden         Web Perlomance       Web Senices Sener für IBM i Programmobjekte Konfiguriet werden konnen. Außerdem werden werden         Be	Allgemeine Tasks und Assistenten 9 Web Services Server erstellen 1 HTTP-Server erstellen 2 Anwendungsserver erstellen 9 WebSphere Portal erstellen 2 WebSphere Portal erstellen	1 ervices Server verwalten Mowson
Server-Eigenschaften b Eigenschaften b Berver-Tatig b HTTP-Server anzeigen b HTP-Server anzeigen b HTP-Server anzeigen b Meb Pafromance Advisor Fehlerbestimmung b Protokolie anzeigen b Meb Day Inother b Web Day I	Web-Service-Assistenten 3 Neuen Service implementieren Web Service	ze Server für Test
Felierbestimmung De Protokolie anzeigen Web Log Montor # Übersicht anzeigen     Server: "DEMOWS01"       Gerwert Temp WegRADD1	Server-Eigenschaften b Eigenschaften b Server-Trading b HTTP-Server anzeigen Services b Implementierte Services verwalten Web Performance Wuch Performance	Ein Web Services Server ist eine komfortable Möglichkeit, vorhandene Programme, die unter IBM i laufen, wie z. B. RPG- und COBOL-Programme, als Web-Services auszulagern. Web-Service-Clients können dann unter Verwendung von Web-Service- basierten Kommunikationsprotokollen nach dem Branchenstandard (z. B. SOAP) mit diesen Services auf IBM i-Programmbasis aus dem Internet oder Intranet interagieren. Die Clients können mittels zahlreicher Platformen und Programmiersprachen wie z. B. C. C++, Java und .NET implementiert werden. Es wird ein benutzenfreundlicher Assistent bereitgestellt, mit dessen Hilfe der Web Services Server und die Services für IBM i-Programmobjekte konfiguriert werden können. Außerdem werden weitere Verwaltungsfunktionen, wie das Starten. Stoppen und Löccher von Services, bereitgestellt. Weitere Informationen erhalten Sie auf folgenden Seiten: http://www-03.ibm.com/systems/i/software/iws/
		Implementierte Services

Implementierte Services verwalten

Mithilfe dieser Auswahl gelangt man in den Übersichtsbereich der definierten Web Services, der als zentrale Administrationseinheit genutzt wird.

Integrierte Web Services

apite

Seite 21

Je nach System kann der Server-Start einen Augenblick lang dauern.

Damit sind die Teilschritte abgeschlossen und der Web Service kann genutzt werden.

Im linken Navigationsbereich der Anzeige findet man den Eintrag "Implementierte Services verwalten"



Kapitel

# Web-Programmierung

### 5.5.3 Seite 22

CHTTP-Server-Administration auf	ASZEBIS - Windows In	ternet Explorer				
💽 🗢 🙋 http://aszebis:2001/HT	TPAdmin				🖌 🗲 🗙 🔁 Bing	<b>ب</b> و
Datei Bearbeiten Ansicht Eavoriten	Extras ?					
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene S	iites 🔹 🙋 Web Slice-Katak	g •				
HTTP-Server-Administration auf ASZEBI	s				🏠 🔹 🔝 🕤 🖃 🌐 👻 Seijte 🕶 S	iigherheit • Extras • 🔞 • 🏾 »
IBM Web Administration for i Konfiguration Verwaltung Erwe	eitert   Zugehörige Links				()	WebSphere.
Alle Server   HTTP-Server Applica	ation Server					
🕒 Gestartet 🕞 🛄 💋 Server: [	DEMOWS01 - V1.3 (We	b-Services) 🔽				
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> <li>HTTP-Server erstellen</li> </ul>	DEMOWS01 > Implement	ntierte Services ver	rwalten alten			
Anwendungsserver erstellen     WebSphere Portal erstellen	Stand vom 13.08.2011 0	7:43:06.				
Veb-Service-Assistenten	Implementierte Services	0				
▼ Server-Eigenschaften	Servicename	Status	Starttyp	WSDL - Servicedefinition		
B Eigenschaften B Server-Tracing	Convert lemp	Gestartet	Automatisch	M Definition anzeigen		
B HTTP-Server anzeigen	O WSGRADUT	Gestallet	Automatisch	1 Definition anzeigen		
✓ Services ⓑ Implementierte Services verwalten	Implementieren	ktualisieren				
✓ Web Performance <sup>©</sup> ) Web Performance Advisor						
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> <li>Web Log Monitor</li> <li>Übersicht anzeigen</li> </ul>	Schließen					
S Contraint an Longon						
< >					Caluda - Takanak	· · · · · · · · · · · · · · · · · · ·
					S Lokales Intranet	🦓 • 🔍 100% •

Die verfügbaren Services





# 5.5.4 Verwaltung der definierten Web Services

In der Anzeige "Implementierte Services verwalten" findet sich die Übersicht der definierten Web Services, die von zentraler Stelle aus verwaltet werden können.

Um einen einzelnen Web Service verwalten zu können, muss er in der Übersicht markiert werden, da sich anschließend die Anzeige ändert. Dort erscheinen Auswahlmöglichkeiten zum Starten bzw. Stoppen des Web Services:

C HTTP-Server-Administration auf	f ASZEBIS - Windows Internet Explorer	
	TTPAdmin 🔽 🗗 🕅 🕅 🖉	0-
Datei Bearbeiten Ansicht Eavoriten	Extras ?	
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene S	Sites 🔹 🙋 Web Silce-Katalog 🔹	
6 HTTP-Server-Administration auf ASZEBI	15 📩 🕆 🔂 - 🗔 👘 - Sejte - Sigherheit - Extras - 🔞	- »
IBM Web Administration for i Konfiguration Verwaltung Erwe	veleri   Zugehörige Links	i.
Alle Server   HTTP-Server Applic	cation Server	
🗣 Gestartet [ 🔲 🛃 🛛 Server:	DEMOWS01 - V1.3 (Web-Services)	
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> <li>HTTP-Server erstellen</li> <li>Anwendungsserver erstellen</li> <li>WebShahere Portal erstellen</li> </ul>	DEMOWS01 > Implementierte Services verwalten Implementierte Services verwalten	
▼ Web-Service-Assistenten <sup>1</sup> Neuen Service implementieren	Stand vom 13.08.2011 07.44.47. Implementierte Services: 💿	
<ul> <li>▼ Server-Eigenschaften</li> <li>B Eigenschaften</li> <li>B Server-Tracing</li> <li>B HTTP-Server anzeigen</li> </ul>	Servicename         Status         Starttyp         WSDL - Servicedefinition           O         ConvertTemp         @Gestantet         Automatisch         @ Definition anzeigen           Image: WSGRAD01         @Gestantet         Automatisch         @ Definition anzeigen	
✓ Services	Implementieren Stoppen Eigenschaften Deinstallieren Aktualisieren Service testen	
✓ Web Performance ♥ Web Performance Advisor		
<ul> <li>▼ Fehlerbestimmung</li> <li>B Protokolle anzeigen</li> <li>4 Web Log Monitor</li> <li>2<sup>9</sup> Übersicht anzeigen</li> </ul>	Schließen	
	े Stokales Intranet 🖓 • 🔍 100%	•

Einzelnen Web Service verwalten

Wie Sie in der folgenden Abbildung sehen können, lassen sich die Web Services von dieser Position aus starten, stoppen, testen und bei Bedarf auch deinstallieren.





Verwaltung der definierten Web Services

5.5.4 Seite 2 Mit der Option "Definition anzeigen" erhält man einen Einblick in das Innenleben des Web Services. Mit seiner Erstellung hat der Assistent die eingegebenen Informationen in XML umgesetzt. Das Ergebnis unseres Beispiels schaut dann wie folgt aus:

C http://aszebis:10054/web/services/WSGRAD01?wsdl - Windows Internet Explorer	
🔆 🔄 🖉 http://aszebis:10054/web/services/WSGRAD01?wsdl	ng 🖉 🖓
Datei Bearbeiten Ansicht Eavoriten Extras ?	
👷 Favoriten 🛛 🖕 🏉 Vorgeschlagene Sites 👻 🖉 Web Slice-Katalog 🔹	
🍘 🕅 🖉 🕲 🖉 🖸	🖶 🔹 Seite 🗸 Sigherheit 🗸 Extras 🗸 🕢 🎽
<pre><d>chml version="1.0" encoding="UTF-8" ?&gt; </d></pre> <ul> <li><vusdl:definitions scap1"="" targetnamespace="http://wsgrad01.wsbeans.iseries" td="" vmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" wsdl="" xml:="" xmlns:axis2="http://wsgrad01.wsbeans.iseries" xmlns:ns="http://wsgrad01.wsbeans.iseries" xmlns:ns0="http://schemas.xmlsoap.org/wsdl/scap1" xmlns:nttp="http://schemas.xmlsoap.org/wsdl/http/" xmlns:xs="http://wsgrad01.wsbeans.iseries" xmls:="" xmls::="" xmls:sinap="http://wsgrad01.wsbeans.iseries" xmls:sinttp="http://schemas.xmlsoap.org/wsdl/http/" xmls:sisses:="" xmls<="" xmlsoap.org=""><td>(2/" "qualified"</td></vusdl:definitions></li></ul>	(2/" "qualified"
<pre><xs:element minoccurs="0" name="_TEMPOUT" nillable="true" type="xs:string"></xs:element></pre>	<u> </u>
Fertig Schales Intra	net 🦓 🔹 🔍 100% 👻 💥

XML des Web Services

Hiermit sind die Konfigurationsschritte für das Bereitstellen eines RPG-Programms als Web Service abgeschlossen.

IBM stellt zudem einen Web Service Test-Client bereit, der automatisch mit der Auswahl "Service testen" gestartet wird.





5.5.4 Seite 3

Verwaltung der definierten Web Services

HTTP-Server-Administration auf A	ASZEBIS - Windows Internet Explorer	
🔆 🔄 💌 🙋 http://aszebis:2001/HTTI	PAdmin	
Datei Bearbeiten Ansicht Eavoriten	Extras ?	
🚖 Favoriten 🛛 👍 🏉 Vorgeschlagene Sit	tes 🔹 💋 Web Slice-Katalog 🔹	
HTTP-Server-Administration auf ASZEBIS		🦄 🔹 🔝 👘 🔹 Seite 🕶 Sigherheit 👻 Extras 🕶 🔞 👻 🎽
IBM Web Administration for i		((?) WebSphere
Konfiguration Verwaltung Erwei	tert   Zugehörige Links	
Alle Server   HTTP-Server   Applicat	tion Server	
🛛 Gestartet 🜓 🛄 🛃 Server: Di	EMOWS01 - V1.3 (Web-Services)	
Allgemeine Tasks und Assistenten	DEMOWS01 > Implementierte Services verwalten	
U HTTP-Server erstellen	Implementierte Services verwalten	
WebSphere Portal erstellen		
Weh-Service-Assistenten	stand vom 13.08.2011 08:05:58.	
Veuen Service implementieren	Implementierte Services: 📀	
<ul> <li>Server-Eigenschaften</li> </ul>	Servicename Status Starttyp WSDL - Servicedefinition	
D Eigenschaften D Server-Tracing	Convertiemp Gestartet Automatisch Model Definition anzeigen	1
B HTTP-Server anzeigen	WSGPAD01     Gestatet Automatisch      Definition anzeigen	
<ul> <li>Services</li> </ul>	Implamentionen Otracen Einenstellen Deinstellingen	
Implementierte Services verwalten	Implementeren Stoppen Eigenschalten Deinstalleren Aktualisieren	Service testen
r Web Performance		
Fehlerbestimmung		
Protokolle anzeigen Web Log Monitor	Schließen	
P Übersicht anzeigen		
		Suckales Intranet 🕢 🗣 🔍 100% 🔹 📑

Service testen

Der Test-Client startet in einem separaten Browserfenster:

🖉 Web Services Explorer - Windows Internet Explorer		
😋 😔 💌 🔊 http://aszebis:2006/wsexplorer/wsexplorer.jsp?	wsdl=http://aszebis:10054/web/services/WSGRAD01?wsd	P •
Datei Bearbeiten Ansicht Favoriten Extras ?		
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene Sites 👻 🙋 Web Slice-Ka	stalog •	
6 Web Services Explorer	🏠 🔻 🔝 🚽 📑 🔫 Sejte 🗸 Sigherheit 🗸	Extras 🔹 🔞 🔹 🎽
i5/0S Web Services Test Client		
😪 Navigator 🔗 📿	Actions	Q
B: WSDL Man         B: Anter://sszebis:10054/neb/services/WSGRAD01/nisdl         B: O: WSGRAD01         B: O: WSGRAD01SO#11Binding         D: Mits://sszebis:0054/neb/services/WSGRAD01/nisdl         B: O: WSGRAD01AD11Binding         D: Mits://sszebis:0054/neb/services/WSGRAD01/nisdl         D: M USGRAD01         M USGRAD01         D: M USG	Invoke a WSDL Operation Enter the parameters of this WSDL operation and dick Go to invoke. Endpoints Endpoints Intro://dezebis:10054/web/services/VSGRADD1 V Content Intro://dezebis:10054/web/services/VSGRADD1 V Solution: Intro://dezebis:10054/web/services/VSGRADD1 V Content Intro://dezebis:10054/web/services/VSGRADD1 V Values Intro://dezebis:10054/web/services/VSGRADD1 V Intro://dezebis:10054/web/services/VSGRADD1 V Values Intro://dezebis:10054/web/services/VSGRADD1 V Values Intro://dezebis:10054/web/services/VSGRADD1 V Intro://deze	Saurce
	Sector States Intranet	🔍 100% 🔹

Ergänzung 19/2011

Web Service Test-Client

Hier wird der Eintrag "converttemp" in der linken Navigationsleiste ausgewählt.





Verwaltung der definierten Web Services

5.5.4 Seite 4 Darauf erfolgt im Hauptfensterbereich die Parameter-Anzeige für den Web Service. Anschließend wird auf "add" geklickt, um die Parameter und deren Inhalte angeben zu können. Mit einem Klick auf "GO" wird dann der Web Service ausgeführt.

Damit ist die Web Service Server-Instanz erstellt. Zudem wurde diesem Server ein Web Service hinzugefügt, wobei ein einzelner Server auch mehrere Web Services zeitgleich bedienen kann. Um weitere Web Services hinzuzufügen, kann man in der Verwaltungsübersicht im Navigationsbereich auf den Eintrag "Neuen Service implementieren" klicken.

CHTTP-Server-Administration au	f ASZEBIS - Windows Inter	net Explorer				
💽 🗢 🙋 http://aszebis:2001/HT	TTPAdmin				🖌 🗲 🗙 🔁 Bing	P •
Datei Bearbeiten Ansicht Favoriten	Extras ?					
🖕 Favoriten 🛛 👍 🏉 Vorgeschlagene :	Sites 🔹 🙋 Web Slice-Katalog 🔹					
HTTP-Server-Administration auf ASZEBI	IS				🟠 • 🖻 · 🗆 🖶 •	Seite + Sigherheit + Extras + 🕢 + 🎽
IBM Web Administration for i Konfiguration Verwaltung Erw	eitert   Zugehörige Links					(?) WebSphere.
Alle Server   HTTP-Server Applic	ation Server					
🛚 Gestartet 下 🗖 🗭 Server:	DEMOWS01 - V1.3 (Web-S	Gervices) 🔽				
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li><sup>™</sup> Web Services Server erstellen</li> <li><sup>™</sup> HTTP-Server erstellen</li> <li><sup>™</sup> Anwendungsserver erstellen</li> <li><sup>™</sup> WebSphere Portal erstellen</li> </ul>	DEMOWS01 > Implementier	te Services verwalte	en en			
▼ Web-Service-Assistenten	Implementierte Services:	0:40.			1	
<ul> <li>Server-Eigenschaften</li> <li>Eigenschaften</li> <li>Server-Tracing</li> <li>HTTP-Server anzeigen</li> </ul>	ConvertTemp WS001	Status Gestartet Wird installiert	Automatisch Automatisch	WSDL - Servicedefinition Definition anzeigen Definition anzeigen Definition anzeigen		
▼ Services	Implementieren Stop	pen Eigensch	naften Deins	stallieren Aktualisieren	Service testen	
▼ Web Performance © Web Performance Advisor						
<ul> <li>▼ Fehlerbestimmung</li> <li>Ď Protokolle anzeigen</li> <li>④ Web Log Monitor</li> <li> <sup>®</sup> Übersicht anzeigen     </li> </ul>	Schließen					
Fertig	9				Schales Intranet	🖓 + 🔍 100% + 🛒

Neuen Service implementieren

Damit wird der Konfigurationsassistent für die Bereitstellung des Web Services gestartet. Die notwendigen Schritte haben Sie bereits kennengelernt.





# 5.5.5 Seite 1

# 5.5.5 Node JS und System i

Performante Webapplikationen, basierend auf der Verwendung von Standards sowie den Funktionen und Möglichkeiten, die das System i bietet? Zugegeben, das klingt paradox – zumindest, wenn es um die von IBM angebotenen Produkte geht, die den System-i-Betreibern in diesem Zusammenhang zur Verfügung stehen.

Seit einiger Zeit steht aber eine Option zur Verfügung, die Anlass zu der Hoffnung gibt, grafische Anwendungen (Webapplikationen) in Verbindung mit dem System i einsetzen zu können. Das war und ist heute natürlich auch schon möglich, aber mit dem Einsatz von Node.JS eröffnen sich hierbei durchaus neue Möglichkeiten.

Node.JS ist eine Open-Source-Anwendung, die neuerdings auch von System-i-Betreibern eingesetzt werden kann, etwa um serverbasierte Java-Script-Anwendungen auszuführen. Node.JS zeichnet zum einen eine schlanke Architektur aus, die eine entsprechend einfache Anwendung nach sich zieht, zum anderen ist es mit der Anwendung möglich, serverbasierte JavaScript-Anwendungen auch ohne einen Browser auszuführen.

Die Verfügbarkeit von Node.JS auf dem System i eröffnet den IBM-i-Betreibern neue Wege in die Welt der browserbasierten Anwendungen oder auch der Anwendungen, die mit Hilfe von JavaScript ausgeführt werden müssen. Node. JS kann auch in Verbindung mit dem Webserver des System i genutzt werden. Dazu liefert IBM im Lizenzprogramm 5733-OPS die notwendigen Funktionen und Werkzeuge. Bevor diese aber zusammen mit dem Webserver eingesetzt werden können, muss er erst noch entsprechend angepasst werden. Diese Anpassung erfolgt über die Implementierung des Plug-in "fastCGI", das sich in der Verzeichnisstruktur der Node.JS-Installation befindet: /QOpenSys/QIBM/ProdData/Node/os400/node-fastcgi



Node JS und System i



Node.JS-Implementierung im IFS

Nachfolgende Abbildung skizziert die Einbindung des fastCGI Plug-in.



#### Skizze fastCGI-Plug-in-Implementierung

Es ist wichtig, die Implementierung in jedem Fall mit entsprechenden Sicherheitseinstellungen abzuschließen, damit keine Sicherheitslücken auf dem System geöffnet werden.

Node.JS ist eine Technologie zur performanten und einfachen Implementierung von Webapplikationen, die auf JavaScript basiert. JavaScript selbst ist im Browser-Umfeld weit verbreitet und kann nun auch in Form von Node.JS zusammen mit Servern effizient eingesetzt werden, und zwar deshalb, weil im Node.JS-Paket bereits eine Fülle an vordefinierten Funktionen enthalten ist, die als Grundlage für Webapplikationen auch zusammen mit dem System i genutzt werden können. Grundlage für die Entwicklung ist die Chrome V8 JavaScript Engine, mit der unter anderem das Thema "Lightweight" berücksichtigt wird. Hier geht es um eine schlanke Form der Verarbeitung, die sich nicht nur durch deren einfachen Einsatz auszeichnet, sondern mit ein wenig Geschick auch performante Lösungen hervorbringt.



Node JS und System i

Node.JS basiert auf dem Einsatz eines Webservers und einiger Konfigurationsschritte. Leider stehen derzeit (Stand 7.2) vonseiten der IBM keine Verwaltungsfunktionen zur Verfügung, mit denen die Serverinstanz sauber verwaltet (zum Beispiel beendet) werden kann. Dieses Nichtvorhandensein lässt sich jedoch mit einigen administrativen Eingriffen bewerkstelligen.

Durch die Verbindung von Node.JS mit dem System i können Webapplikationen zusammen mit dem Betriebssystem IBM i eingesetzt werden. Mit ein wenig Geschick lassen sich hier RPG-Programme verwenden, SQL-Funktionen nutzen oder beispielsweise auch IBM-i-Befehle absetzen.

Die Implementierung von Node.JS basiert auf den folgenden Teilbereichen:

- Basis-Node.JS-Funktionen,
- DB2-Zugriffen und
- der IBM-i-Objektverwendung.

Zusammen mit der Node.JS-Funktionalität liefert IBM unter anderem auch ein API aus, mit dem direkt auf DB2-Komponenten zugegriffen werden kann. Allerdings dürfen dabei Sicherheitsaspekte nicht außer Acht gelassen werden. So berücksichtigt die Node.JS-Anwendung zum Beispiel den Benutzer, mit dem der Webserver gestartet wurde. Wenn die Sicherheitsaspekte in einem Konzept ausreichend bedacht wurden, dann kann die Kombination aus Auslieferung eines API und Sicherheitsaspekte als Grundlage für eine Webapplikation genutzt werden. Im Gesamtkontext bietet Node.JS zusammen mit der DB2 for i access library eine Option dafür, Webapplikationen zu schreiben, die einen direkten DB2-Zugriff ermöglichen. DB2 for i access library ist für JavaScript geschrieben und erlaubt unterschiedlichste Arbeiten mit der DB2/UDB des System i. SQL-Anweisungen lassen sich so direkt ausführen, und in Verbindung mit JSON ist die Ergebnisverarbeitung möglich.

IBM ist dazu übergegangen, einige Open-Source-Komponenten für das System i in Form eines speziellen Lizenzprogramms (5733OPS) den Kunden zur Verfügung zu stellen. Dieses Lizenzprogramm ist zwar kostenfrei, bedarf aber nach der Installation im Regelfall noch einer Komplettierung durch spezielle PTFs. In diesem Fall muss auf dem System i das aktuelle Gruppen-PTF für den http-Server installiert werden.

So wird unter anderem der XMLSERVICE bereitgestellt – ein weiteres Open-Source-Produkt, das in die Node.JS-Umgebung eingebunden ist.

Infolge der sicher noch anstehenden Entwicklungen und Erweiterungen darf basierend auf der jetzigen Version von Node.JS auf weitere Neuerungen gehofft werden.

Ergänzung 38/2016





Node JS und System i

5.5.5 Seite 4 Die Node.JS-Spezifika für das System i beziehen sich auf die Arbeit mit den DB2-Objekten des System i sowie die Nutzung zusammen mit der IBM-i-Umgebung und der Implementierung in die bestehende Anwendungsperipherie. Leider unterstützt Node.JS keine EBCDIC-Zeichensätze – also auch nicht den Zeichensatz, der per Standard auf dem System i verwendet wird. Achten Sie deshalb darauf, dass Sie die korrekte Zeichensatzeinstellung verwenden, damit die Node.JS-Implementierung auch fehlerfrei funktioniert. Es empfiehlt sich im Allgemeinen, UTF-8 zu nutzen, um Probleme, die im Zeichensatz begründet liegen, zu vermeiden.

Wenn Sie Node.JS einsetzen wollen, dann benötigen Sie dazu nachfolgende Software-Voraussetzungen in Form der Lizenzprogramme:

•	5733OPS	Option 1	IBM i Open Source Solutions
---	---------	----------	-----------------------------

- 5770SS1 Option 33 PASE
- 5733SC1 Option 1 OpenSSH, OpenSSL
- 5770DG1 \*BASE IBM I WebServer

Ausgehend davon, dass im Regelfall das Lizenzprogramm 5733OPS auf dem System nicht installiert ist, muss dies meist nachgeholt werden. Das noch recht neue Lizenzprogramm 5733OPS steht seit der Betriebssystemversion 7.1 zur Verfügung.

Die Installation ist wie folgt durchzuführen:

- RSTLICPGM LICPGM(5733OPS) DEV(Bandlaufwerk) OPTION(\*BASE) LNG(2924) RLS(V1R1M0)
- RSTLICPGM LICPGM(5733OPS) DEV(Bandlaufwerk) OPTION(1) LNG(2924) RLS(V1R1M0)



Ein Beispiel für RSTLICPGM



Node JS und System i



Nach der Installation der erforderlichen Lizenzprogramme müssen die folgenden PTF-Gruppen auf dem System vorhanden sein bzw. installiert werden.

Für Version IBM i 7.1:

- SF99368 mindestens Level 31
- SI55522 (für Lizenzprogramm 5733SC1)
- SI50835 (für XMLService Toolkit)

Für Version IBM i 7.2:

- SF99713 mindestens Level 5
- SI51153 (für XMLService Toolkit)

Sind diese Voraussetzungen erfüllt, dann finden Sie im IFS Ihres Systems unter anderem auch das folgende Verzeichnis: /QOpenSys/QIBM/ProdData/Node/

Die korrekte Installation kann mit Hilfe der nachfolgenden Teilschritte verifiziert werden:

1. Erweitern Sie die QShell-Umgebung durch das Hinzufügen der Umgebungsvariablen QIBM\_MULTI\_THREADED. Dazu geben Sie in einer IBM-i-Befehlszeile den Befehl ADDENVVAR ENVVAR(QIBM\_MULTI\_ THREADED) VALUE(Y) ein.

tā tē up communita šāms tātus vēkus Na Alba ar albā	
Image (1900)       Por (2)         WeakEddow D       Encode    Umgebungsvariable hinzufügen (ADDENVVAR)          Auswahl eingeben und Eingabetaste drücken.         Umgebungsvariable	
Heg [1320/// Pot [3]     Wordshow D       Umgebungsvariable hinzufügen (ADDENVVAR)       Auswahl eingeben und Eingabetaste drücken.       Umgebungsvariable	
Umgebungsvariable hinzufügen (ADDENVVAR) Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable <u>QIBH MULTI THRREDED</u>	
Umgebungsvariable hinzufügen (ADDENVVAR) Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable	
Umgebungsvariable hinzufügen (ADDENVVAR) Auswahl eingeben und Eingebetaste drücken. Umgebungsvariable <u>QIBH MULTI_THRREDED</u>	
Umgebungsvariable hinzufügen (ADDENVVAR) Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable	
Umgebungsvariable hinzufügen (ADDENVVAR) Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable	
Umgebungsvariable hinzufügen (ADDENVVAR) Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable <u>QIBM MULTI_THRAEDED</u>	
Umgebungsvariable hinzufügen (ADDENVVRR) Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable	
Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable <u>QIBM MULTI THRAEDED</u>	
Auswahl eingeben und Eingabetaste drücken. Umgebungsvariable <u>QIBH_MULTI_THRREDED</u>	
Umgebungsvariable <u>QIBH_MULTI_THRAEDED</u>	
Umgebungsvariable <u>QIBH.HULTI_THRAEDED</u>	
Umgebungsvariable <u>QIBM_MULTI_THRREDED</u>	
Anfangswert	
Ende	
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F10=Zusätzl. Parameter	
riz-Hobrechen ris-verwendung der Hizeige rz4-wertere rasten	
C 08/037	
202 - Session successfully started	



**Ein ITP Handbuch** 

Node JS und System i

#### 5.5.5 Seite 6

Sollten Sie diese Einstellung bereits gesetzt haben, dann können Sie diesen Schritt auch überspringen.

- 2. Starten Sie die Shell-Umgebung. Geben Sie dazu in der Befehlszeile den Befehl QSH ein.
- 3. Führen Sie in der Shell-Umgebung den folgenden Befehl zur Überprüfung der installierten Node.JS-Umgebung aus: node –v
- 4. Als Ergebnis müssen Sie eine Nachricht erhalten, aus der die installierte Node.JS-Version hervorgeht. Ein Beispiel dazu zeigt nachfolgende Abbildung:

>	node -v
	v0.10.29
	\$
>	npm -v
	1.4.14
	\$

#### Fertig-Meldung

Wenn Sie diese Anzeige erhalten, dann ist das System i bereit, Node.JS zu unterstützen.

Geben Sie nun den Befehl "node" ein und starten Sie damit die interaktive Umgebung für Node.JS, die auch Node.JS REPL genannt wird.

Im Folgenden finden Sie eine erste einfache Anwendung, mit der Sie die Ausführung von Node.JS exemplarisch angehen können. Dazu eignet sich am besten das berühmt-berüchtigte "Hello World".

Erstellen Sie den nachfolgend gezeigten Codeausschnitt in einer Datei im IFS. Beispielhaft wird dazu das Verzeichnis/home/zeig verwendet. Die Datei wird exemplarisch "Beispiel1.js" genannt.

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(8080, '192.168.178.40');
console.log('Server auf http://192.168.178.40:8080/');
```



Node JS und System i



Seite 7



Codebeispiel "Beispiel1.js"

Achten Sie darauf, dass Sie die beiden Variablen für die Adresse und den Port durch die Spezifika Ihres Servers ersetzen, dass Sie also die IP-Adresse und den Port Ihres Systems/Servers verwenden.

Um den Webserver zu starten, führen Sie nach dem Speichern der Datei das Programm aus. Dazu starten Sie eine QSH-Sitzung und wechseln dort zunächst in das Verzeichnis, in dem sich das Programm befindet. Cd/home/zeig/



#### Das Verzeichnis wechseln

Anschließend rufen Sie den Programmcode auf: Node Beispiel1.js

Anstelle der beiden Einzelaufrufe können Sie auch den folgenden qualifizierten Aufruf verwenden: Node/verzeichnisname/njs/Beispiel1.js





Node JS und System i

### 5.5.5 Seite 8

Jetzt können Sie eine Browser-Sitzung starten. Dazu sollte auf dem angegebenen Webserver die Anwendung "Hello World" laufen und aufrufbar sein. Um dies zu testen, geben Sie im URL-Eingabebereich des Browsers die Serveradresse und den Port an, den Sie zuvor in der Konfigurationsdatei verwendet haben. Im Beispiel wird der Port 8080 benutzt, der auch in der Datei verwendet wurde.

http://192.168.178.40:8080	+
€ € 192.168.178.40:8080	⊽ C <sup>i</sup>
Hello World	

Ergebnis-Anzeige im Webbrowser

Wenn die "Hello World"-Anzeige erscheint, dann ist die Basiseinrichtung des Webservers so weit abgeschlossen, dass über den Port 8080 die Anfrage und auch die Antwort verarbeitet werden können.

Auf diese Weise wurde das Programm "Beispiel1" aufgerufen, und gleichzeitig wird der Webserver gestartet. Das, obwohl dieser zuvor nicht aufwändig konfiguriert werden musste!

Sollten Sie sich bereits in der Vergangenheit mit anderen Möglichkeiten des Ausführens von Webapplikationen zusammen mit dem System i beschäftigt haben, dann werden Sie mir sicher zustimmen: Die Verwendung von Node.JS ist mit den bisherigen Implementierungen von Webapplikationen nicht zu vergleichen und zudem wesentlich einfacher in der Verwaltung und Handhabung. Und das ist noch nicht alles. Lesen Sie weiter.

Als Erstes genügt ein Blick in den Programmcode.

- var http = require(,http');
   Mit der Anweisung var http = require(,http'); werden die http-Funktionen bereitgestellt, die das Programm nutzen soll.
- http.createServer(function (req, res) Die Anweisung "createServer" erstellt einen Webserver. Details wie Serveradresse und Port sind in den folgenden Parametern hinterlegt. Wenn Server und Anwendungen auf einem System i liegen, dann muss die IP-Adresse nicht zwingend eingegeben werden. In diesem Fall ist lediglich der Port, gefolgt von 0.0.0.0, anzugeben. Damit werden die LOCALHOST-Einstellungen verwendet. Beispiel: listen(8080, ,0.0.0.0<sup>c</sup>);

Dieses erste einfache Beispiel soll nur die Lauffähigkeit des Node.JS-Servers illustrieren. Alternativ zur Verwendung von QShell oder Putty können auch andere geeignete Werkzeuge genutzt werden. Da Node.JS eng mit Google verbunden ist, kommt hier wohl ein "Exote" infrage: Secure Shell Chrome.



Node JS und System i

Ist diese einfache Anwendung lauffähig, dann können auch "wirkliche Anwendungen" entwickelt und genutzt werden. Anhand eines einfachen Beispiels soll hier gezeigt werden, wie das API "DB2 for i access library" zusammen mit Node. JS verwendet werden kann, um auf die Inhalte der DB2 zugreifen zu können.

Zunächst muss sicherheitshalber geprüft werden, ob das API auch auf dem System installiert wurde. Dazu muss folgendes Verzeichnis gesucht werden: /QOpenSys/QIBM/ProdData/Node/os400/db2i/.

Die Bereitstellung der API-Funktionen ist denkbar einfach – es muss nur der folgende Eintrag in den Quellcode eingebunden werden: var db = require('/QOpenSys/QIBM/ProdData/Node/os400/db2i/lib/db2');

Das Vorhandensein der erforderlichen Komponenten kann entsprechend geprüft werden – gemäß dem zuvor genannten Pfad muss dort der Eintrag db2. js existieren.





Sobald dieser Eintrag in den Quellcode eingebunden wurde, können, um mit dem Inhalt der Datenbank arbeiten zu können, herkömmliche SQL-Befehle genutzt werden.





Seite 10

Node JS und System i

**5.5** Dazu folgendes Beispiel:

Mittels einer SQL-Select-Anweisung wird der Inhalt der Datei KUNDEN abgerufen.

Beim Codieren sind einige Punkte zu beachten, die Ihnen möglicherweise aus der RPG-Programmierung bekannt vorkommen:

- Bereitstellen der notwendigen Funktionen, was in etwa einer aus RPG bekannten Copy-Strecke gleichkommt. Für die Verwendung der DB2-Funktionen erfolgt die Bereitstellung mit der folgenden Codezeile: var db = require(,/QOpenSys/QIBM/ProdData/Node/os400/db2i/lib/db2<sup>c</sup>);
- 2. Die Umgebung sollte initialisiert werden. Dazu verwenden Sie die Anweisung db.init().
- 3. Es muss eine Verbindung zur Datenbank hergestellt werden. Dazu verwenden Sie die Anweisung db.conn.
- 4. Die Anweisungen codieren Sie im Abschnitt db.exec().
- 5. Die Ausgabe wird über einen Parameter gesteuert. Im Beispiel erfolgt die Ergebnisanzeige auf der Konsole. Dazu nutzen Sie den Eintrag console.log. Es hängt von den möglichen Ergebniswerten ab, wie das Ergebnis angezeigt wird. Entscheidend ist hier die Tatsache, ob ein Ergebnis oder eine Liste von Ergebnissen geliefert werden soll.
- 6. Das Ende markieren Sie (db.close).

Damit ist der notwendige Funktionsumfang abgeschlossen. Hier handelt es sich natürlich nur um die grobe Struktur – die Funktionalität wird sich in der Anwendung in verschiedenen Ausprägungen der Anweisungen, insbesondere im Bereich db.exec, finden.

Das Script-Beispiel ist denkbar einfach. Es wird davon ausgegangen, dass sich sowohl der Webserver als auch die DB2-Datenbank auf demselben System befinden. Allerdings ist das kein Muss – das Script könnte auch so geschrieben werden, dass auf eine ferne Datenbank (DB2) zugegriffen wird. Hier soll aber erst einmal das einfachste DB2-Beispiel genutzt werden.



Node JS und System i



Seite 11





Codebeispiel für DB2-Zugriff und SQL-Verwendung

Im Beispiel wird zunächst die Umgebung initialisiert. Dazu wird der Eintrag db.init() genutzt. Diese Anweisung muss zwingend vor der Conn-Anweisung codiert werden.

Anschließend wird mit der Anweisung db\_conn () eine Verbindung zur DB2-Datenbank hergestellt. In Abhängigkeit vom Servermodus (Servermodus oder Nichtservermodus) müssen hier neben dem Datenbanknamen auch der IBM-i-Benutzer und dessen Kennwort angegeben werden. Den DB2-Datenbanknamen können Sie im Befehl WRKRDBDIRE ermitteln. Wenn der Webserver auf demselben System läuft, auf dem sich auch die DB2-Datenbank befindet, dann können Sie hier den Sonderwert "LOCAL" verwenden.



Node JS und System i

## 5.5.5 Seite 12

Im Programmcode können Sie nun jede Form von SQL-Anweisung einbinden. Dabei greifen natürlich die Berechtigungsprüfungen, die auch nativ auf dem System i durchgeführt werden.

In der Anweisung db.exec wird die SQL-Anweisung codiert. Im Beispiel ist das die SELECT-Anweisung für den Zugriff auf die Tabelle KUNDEN.

Das Ergebnis der SELECT-Anweisung, also die Anzeige der beiden Spalten KDKDNR (Kundennummer) und KDNAME (Kundenname), wird mittels JSON ausgegeben. Es handelt sich also um einen Standard, der in Webapplikationen recht leicht implementiert werden kann.

Nachdem hier die Option des DB2-Zugriffs mittels JavaScript im Groben vermittelt wurde, sollen im nächsten Abschnitt die Möglichkeiten aufgezeigt werden, die sich im Zusammenhang mit der Verwendung von System-i-Objekten bieten.

Eine weitere Säule der Node.JS-Implementierung ist die Option, direkt mit System-i-Objekten zu arbeiten. Die dazu erforderlichen Funktionen sind im Node. JS Toolkit enthalten und basieren auf den bereits früher verfügbaren XMLSER-VICE-Funktionen. Derzeit liefert IBM mit dem Node.JS Toolkit den folgenden Funktionsumfang aus:

itoolkit

Basisfunktionen und Schnittstellen für den Aufruf von

- CL-Befehlen,
- Programmen,
- SQL-Anweisungen
- iwork Funktionen für die Ermittlung von Job- und Systeminformationen
- iprod
   Funktionen f
  ür das Abrufen von Lizenzprogramm- und
   PTF-Informationen
- iuserspace Funktionen für das Arbeiten mit Userspaces
- inetwork
   Funktionen f
  ür das Arbeiten mit Netzwerkinformationen
- iobj
   Funktionen f
  ür die Arbeit mit Berechtigungen, Befehlen und Programmen
- idataq
   Funktionen f
  ür das Arbeiten mit DTAQs



Node JS und System i



Die derzeit verfügbaren Funktionen erlauben einen gewissen Einsatz zusammen mit den nativen System-i-Objekten, womit sich in Webapplikationen durchaus auch komplexere Anwendungen erstellen lassen, mit denen das Arbeiten auf dem System i gesteuert werden kann. Auch wenn nicht alle Funktionsbereiche abgedeckt sind, so lassen sich doch gewisse Anforderungen mit Hilfe der aufrufbaren Programme (im itoolkit enthalten) umsetzen. Im Folgenden wird gezeigt, wie die IBM-i-Objekte mit Node.JS verwendet werden können.

Die Implementierung der Basisfunktionen erfolgt über das bereits erwähnte itoolkit. Diese muss im Script eingebunden sein.

Die Befehlsverarbeitung wird mit allen notwendigen Berechtigungsprüfungen auf dem System i ausgeführt. Die möglichen Ergebnisse werden in JSON zurückgegeben und können dann entsprechend aufbereitet verarbeitet werden.

Diese Einbindung kann mit der nachfolgenden Anweisung durchgeführt werden:

```
var xt = require('/Q0penSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit');
var http = require('http');
var fs = require('fs');
var url = require('url');
var db = require('/QOpenSys/QIBM/ProdData/Node/os400/db2i/lib/db2');
var xt = require('/Q0penSys/QIBM/ProdData/Node/os400/xstoolkit/lib/itoolkit');
var
      DBname = "*LOCAL ";
var ip = "IP_Address";
var port = Port;
var webserver = http.createServer(function (req, res) {
   var realPath = __dirname + url.parse(req.url).pathname;
   fs.exists(realPath,function(exists){
      if(!exists){
         var sql = url.parse(req.url, true).query.sql;
                     var cl = url.parse(req.url, true).query.cl;
         res.writeHead(200, {'Content-Type': 'text/plain'});
         if(sql && sql.length > 0) {
            console.log("SQL statement : " + sql);
            db.init();
            db.conn(DBname);
            db.exec(sql, function(rs) {
               res.write(JSON.stringify(rs));
            });
            db.close():
         3
```



Hier sehen Sie den Code als Ausschnitt im RDI.

Datei Bearbeiten Quelle Refactoring Navigieren Suchen Projekt Ausführen Eenster Hilfe

If Forme Systems 12 9- Team		- F
Arg reme systeme as a ream		
▲ 웹 수 수 졛 ⊟ 역 <u>`</u>	<pre>var xt = require('/gopensys/gibm/Producta/Node/05400/xstoolkit/lib/itoolkit');</pre>	-
TRDRR 🔺	var nutp = require('nutp');	
C TRDSZ	var is = require(13);	
C TRDZG	var die - require( dr);	
webservices	<pre>var ub = require( /gopensys/gibs/Floodback/Node/osto//dbz//lib/ubz/); var ub = require('/gopensys/gibs/Floodback/Node/osto//wit/lib/itoolkit');</pre>	
🗁 ZEIG		
🗀 ch	var DBname = "*LOCAL ":	
🗀 test	<pre>var ip = "IP Address";</pre>	
🗀 XML	var port = Port;	
🗀 Zeig Stammdaten		
🗀 Zeig-Preise	⊖var webserver = http.createServer(function (req, res) {	
🗀 Zeig-QM	<pre>var realPath = dirname + url.parse(req.url).pathname;</pre>	
🗀 Zeig-SKU	fs.exists(realPath, function(exists){	
1252	if(!exists){	-
🐨 4 - XML.doc	<pre>var sql = url.parse(req.url, true).query.sql;</pre>	=
5 SOA Standards.doc	<pre>var cl = url.parse(req.url, true).query.cl;</pre>	
💼 6 Einführung in XPath.doc 🗉	res.writeHead(200, {'Content-Type': 'text/plain'});	
	if(sql && sql.length > 0) {	
Beispiel Word Dokument zeigt den IFS I	<pre>console.log("SQL statement : " + sql);</pre>	
Beispiel Word Dokument zeigt den IFS I	db.init();	
Beispiel Word Dokument zeigt den IFS I	db.conn(DBname);	
Beispiel1.js	db.exec(sq1, function(rs) (	
DB2Script01.js	<pre>ite(usow.stringity(rs)); }</pre>	
x demo1.xml	dh close():	
iToolKitDemo01.js	,	
jcrcmds.bt	$1f(c) & \delta & cl.length > 0)$	
krb5ccname	{	
x mappe2.xml	<pre>console.log("CL statement : " + cl);</pre>	
The second secon	<pre>var conn = new xt.iConn(DBname);</pre>	
	<pre>conn.add(xt.iSh("system -i " + cl));</pre>	
🔲 Eigenschaften 🛛 🔍 🧏 Ferner Arbeitspuffer 🛛 👘 🗖	function cb(str) {	
	res.write(xt.xmlToJson(str)[0].data);	

Codebeispiel im RDI



Wird dieses Beispiel in einem Browser aufgerufen, dann erh Anzeige:	alten Sie folgende
Node.JS Beispiel +	
♦ ♦ Mttp://192.168.178.40:8080/beispiel1.html	
SQL Command	
SELECT * FROM	Query
CL Command	
DSPJOB	Run

Ergebnis-Anzeige im Browser

Es gibt zwei Eingabebereiche: den ersten für die Eingabe und die Ausführung einer SQL-Anweisung, den zweiten für die Eingabe eines CL-Befehls.

Bitte beachten Sie, dass sich hier nicht nur Batch-Befehle eingeben und ausführen lassen, sondern auch Dialogbefehle, wie zum Beispiel die Anzeige eines Jobs. Exemplarisch wird der Befehl DSPJOB verwendet.

Nach einem Klick auf die Schaltfläche "Run" werden Ihnen die Ergebnisse nach der Befehlsausführung des IBM-i-Befehls im Browser angezeigt.

Sollten Sie bei der Ausführung Probleme in der Darstellung der Zeichen oder gar Abbrüche erhalten, dann kann eine Anpassung der CCSID für den Job sinnvoll sein. Das Problem ist meist hausgemacht: Viele System-i-Umgebungen werden mit der CCSID 65535 betrieben. Damit verbunden ist die Zeichenumsetzung. Das heißt, alle Zeichen werden, so wie sie sind, verarbeitet, und es wird eben nicht zwischen den verschiedenen Zeichensätzen konvertiert – ein in internationalen Umgebungen oder solchen, bei denen Webapplikationen zum Einsatz kommen, vermehrt auftretendes Problem. Dieses kann umgangen oder auch beseitigt werden, indem die CCSID geändert wird. Die CCSID kann auf unterschiedlichen Ebenen eingestellt werden. Die höchste ist der Systemwert QCCSID, gefolgt von der CCSIS des Webservers, der CCSID auf der Ebene des IBM-i-Benutzerprofils bis hin zur CCSID auf der Ebene des Jobs.

Sollten Sie Zeichenumsetzungsprobleme beim Einsatz von Node.JS haben, dann prüfen Sie die CCSID und passen diese bei Bedarf an. Nehmen Sie als Beispiel die CCSID-Anpassung auf der Ebene des Jobs: CHGJOB JOB(426569/ZEIG/QPADEV000C) CCSID(273)

Node JS und System i



Seite 15

Node JS und System i





#### CHGJOB

Der Nachteil dieses Verfahrens ist, dass vonseiten der IBM keine Funktion bereitgestellt wurde, mit Hilfe derer der Server sauber beendet werden kann. Sollte es eine Situation geben, bei der ein Job in diesem Bereich beendet werden muss, dann bleibt nichts anderes übrig, als den Webserver bei Bedarf hart zu beenden. Das kann mit dem Befehl WRKACTJOB und der Auswahl 4 auf dem betreffenden Job erfolgen.





## 5.5.6 Seite 1

# 5.5.6 REST-Webservices

Webservices haben sich in den letzten Jahren auf dem System i etabliert und bilden in immer mehr Unternehmen eine wertvolle Schnittstelle zwischen den Systemwelten.

Bis Anfang 2016 war den IBM-i-Betreibern die Nutzung der SOAP-basierten Webservices vorgeschrieben. SOAP (Simple Object Access Protocol) kam dabei in Verbindung mit IBM-i-basierten Programmen oder Serviceprogrammen zum Einsatz. Allerdings kennt die Webwelt nicht nur die SOAP-Webservices, sondern eben auch eine Alternative: die REST-Webservices. Webservices haben die Eigenschaft der Maschine-zu-Maschine-Kommunikation.

Letztlich basieren alle Webservices auf Standards, die mit den neuesten Betriebssystemen genutzt werden können. Der Webstandard HTML ist die Basis für den Einsatz der REST-Webservices. REST wurde erstmals im Jahr 2000 praktisch eingesetzt und hat sich seitdem immer weiter entwickelt. REST ist die Abkürzung für "Representational State Transfer" und ist im Gegensatz zu SOAP eine einfachere Struktur – auch wenn SOAP vom Begriff her ("simple") eigentlich genau das Gegenteil zu vermitteln scheint. Denn mit den Jahren wurden aus den einfachen Strukturen immer komplexere SOAP-Konstrukte.

REST hat den Vorteil, dass seine Architektur eine im Vergleich zu manchen komplexen SOAP-Schnittstellen einfachere Anwendung und Verarbeitung erlaubt. Man geht hier quasi wieder in die Zeiten zurück, als Webanwendungen im Wesentlichen schlanke und performante Anwendungen waren. Basierend auf asynchronem JavaScript und Ajax können so Anwendungen ohne Bezug zu einer spezifischen Middleware erstellt werden.

REST basiert auf HTML-Standards und nutzt zudem XML und auch JSON. HTML-Kenner haben es leicht, die REST-Funktionen zu nutzen, denn die REST-Anforderungen werden mit einfachen HTML-Methoden umgesetzt. Damit verbunden sind deren einfache Implementierung und performante Nutzung der REST. REST-Webservices basieren auf der Grundlage, dass der Service eindeutig über die URI identifiziert werden kann.

**REST-Webservices** 

5.5.6 Seite 2 Die Fragestellung, wann REST und wann SOAP genutzt werden soll, lässt sich nicht global beantworten. Hier finden sich einige Fragestellungen, die im Hinblick auf einen REST-Webservice von Bedeutung sind:

- Wie ist der Aufbau der URI?
- Welche Datentypen sind zu verarbeiten?
- Welche HTTP-Methoden können zugeordnet/genutzt werden?

Folgende HTML-Komponenten stellt REST für den Einsatz mit dem System i bereit:

- POST wird für die Bereitstellung von Ressourcen auf einem Server eingesetzt.
- GET dient dem Auflisten und Abrufen von Ressourcen.
- PUT wird für das Ändern von Ressourcen eingesetzt.
- DELETE dient dem Löschen von Ressourcen.

Ein vager Vergleich zwischen SOAP- und REST-Webservices zeigt, dass Letztere eine Ähnlichkeit mit den SQL-Befehlen INSERT, SELECT, UPDATE und DELTE haben. Demzufolge kommt REST als direkter Methodenaufruf dem Konzept der SQL-Anwendung auf IBM i recht nahe.

Es kommt letztlich auf die Anforderung an, was mit einem Webservice umgesetzt werden soll – hiernach richtet sich auch die Verwendung der HTML-Komponenten. Sollen zum Beispiel lediglich Daten gelesen werden, dann wird die Entscheidung auf die Nutzung der GET-Methoden im HTML-Umfeld fallen.

Jede REST-Nachricht beinhaltet alle für die Kommunikation zwischen Client und Server notwendigen Informationen.

Diese Hauptbereiche lassen sich mit URL-Spezifikationen kombiniert für die Umsetzung der Anforderungen nutzen. Eine weitere wichtige Eigenschaft von REST ist, dass alle Ressourcen, die mit REST-Services eingesetzt werden, eine eindeutige Adresse haben – also eine URL. REST nutzt die URL für die Kommunikation zwischen dem Server und den Clients. Der integrierte Webservice-Webserver nutzt die URLs in Verbindung mit dem REST und übergibt die HTTP-Anforderungen an die entsprechende Aufrufstelle. Nachfolgend finden Sie ein Beispiel für eine solche Aufruf-URL:



http://<host>:<port>/<context-root>/<root-resource>/
<uri-path-template>
**C** Kapitel

Seite 3

Der Aufbau der URL ist im Falle eines IBM-i-Webservices wie folgt:

- Host,
- IP-Adresse oder Host-Name des System i,
- Port,
- Port, der für die Ausführung des Webservices vorgesehen und angegeben ist,
- Context Root,
- per Standard immer mit dem IFS-Pfad /web/services" gefüllt; kann bei Bedarf anhand der Servereigenschaften geändert werden,
- Root Resource,
- Name des Webservices,
- URI Path Template,
- beinhaltet URI-Informationen und definiert die Zuordnung der Ressourcen. Hier können zum Beispiel auch Variablen eingebunden werden. Sie sollten darauf achten, alles in Kleinbuchstaben anzugeben! Zudem haben Sie die Möglichkeit, auf Fehler zu reagieren. Sicher kennen Sie den berühmten 404-Fehler (nicht gefunden). Dieser Fehler-"Hinweis" sagt nichts aus. Sie haben die Option, solche Fehler umzuleiten und eine alternative Webseite im Fehlerfall darzustellen, was ein wenig mit MONMSG vergleichbar ist. Ein Beispiel für eine URI kann wie folgt aussehen: /web/services/kunde/001683

Die Root Source des REST-Webservices bildet das IBM-i-Programm respektive das Serviceprogramm. Das RPG-Programm verfügt über einen Einstiegspunkt, der innerhalb der REST-Services als sogenannte Ressourcenmethode genutzt wird. Eine solche Ressourcenmethode wird direkt der HTTP-Methode zugeordnet.

Hinsichtlich dieses Funktionsumfangs besteht die Möglichkeit, eine Vielzahl an Anforderungen basierend auf REST-Standards mit System-i-Anwendungen abzudecken. Dabei muss insbesondere bei Webanwendungen die Ausführungsgeschwindigkeit beachtet werden. Wenn beispielsweise eine Webanwendung durchgeführt werden soll, dann wird mit einem Mausklick eine unverzügliche Reaktion erwartet. Das muss natürlich wie bei jeder Form der Anwendungsentwicklung beachtet werden. REST bietet auch die Möglichkeit der Lastverteilung durch Cluster-Optionen. Diese erlauben die Verteilung der REST-Dienste auf unterschiedliche Serverinstanzen innerhalb einer Cluster-Umgebung.

Nun soll der REST-Service in Verbindung mit dem System i genauer betrachtet werden.

Zunächst die Voraussetzungen: REST-Webservices sind mit dem Betriebssystem IBM i ab der Version 7.1 (PTF-Stand SF99368 Mindeststufe 32) oder der Version 7.2 (PTF-Stand SF99713 Mindeststufe 6) verfügbar.





5.5.6 Seite 4

### Ressourcenmethode

Wie bereits erwähnt, stellen die Ressourcenmethoden die Verbindung zu den HTTP-Methoden dar, womit sie vergleichbar sind mit dem Einstiegspunkt innerhalb des RPG-Programms. In Anlehnung daran werden mit dem integrierten Webservice-Server des System i folgende Methoden unterstützt:

- DELETE,
- GET,
- POST und
- PUT.

Diese Methoden werden mit den entsprechenden Komponenten auf der HTML-Seite verbunden. Dabei muss darauf geachtet werden, dass die Inhaltsangabe der Ressourcenmethode für den Client definiert wird. Hier können wahlweise folgende Content-Angaben vorgenommen werden:

- \*ALL Content-Typ \*/\*
- \*XML Content-Typ "application/xml"
- \*JSON Content-Typ "application/json"
- \*XML\_AND\_JSON Content-Typen "application/json" und "application/xml"

Achten Sie bei der Entwicklung des Webservices auf die korrekte Zuordnung der Content-Typen. Andernfalls kann es beim Aufruf des Webservices zu einem HTTP-Fehler kommen, und der Webservice bricht dann möglicherweise ab.

Wurde der Content-Typ festgelegt, dann folgt die Angabe der Parameter. Diese sind analog den Parametern zu sehen, die das zugehörige Programm (RPG) erfordert. Auch hier kann zwischen Eingabe- und Ausgabeparametern unterschieden werden.

Basierend auf den weiterführenden URI-Informationen, bietet sich die Option, gewisse Funktionen oder Prozeduren innerhalb eines RPG-Programms direkt zu nutzen.

Die variable Nutzung von XML oder JSON in Verbindung mit den Ausgabeparametern erlaubt einen individuellen Einsatz der REST-Webservices, sorgt aber hin und wieder für Verwirrung. Denn wann kann welche der beiden Varianten verwendet werden?

Die Antwort ist sicher nicht einfach generell zu erteilen. Deshalb sollen hier einige Beispiele/Unterschiede zwischen der XML- und der JSON-Variante aufgezeigt werden.





Nachfolgend finden Sie ein einfaches Beispiel für einen Eingabeparameter. In diesem Beispiel wird basierend auf dem Eingabeparameter "Kundennummer" das Umsatzergebnis des Kunden auf dem System i mit dem Webservice gelesen.

#### <LESKUNDEInput> <KDNRIN>0001608</KDNRIN> </LESKUNDEInput>

Es handelt sich hier um die Wrapped-Variante der Parameterverwendung.

Nehmen Sie einmal an, dass Sie neben der Kundennummer einen Kundenund einen Vertriebsbereich mit angeben können. Bei beiden Parametern handelt es sich um numerische Werte. Das RPG-Programm liefert basierend auf den eingegebenen Parametern das gewünschte Ergebnis und die Umsatzwerte für den eingegebenen Bereich.

Jetzt kann die Eingabe der Parameter in Gänze in XML wie folgt aussehen:

```
<LESKUNDEInput>
<KDNRIN>0001608</KDNRIN>
<STRUKT1>
<FLD1>001/FLD1>
<FLD2>555/FLD2>
</STRUKT1>
</LESKUNDEInput>
```

Bei diesem Beispiel handelt es sich um die Definitionsvariante mit einer Struktur. Die Struktur STRUK1 beinhaltet die beiden Parameter FLD1 und FLD2.

Das XML-Ergebnis könnte als Ergebnis folgendes XML liefern:

```
<LESKUNDEErgebnis>
<UMSATZ>16722</UMSATZ>
</LESKUNDEErgebnis>
```

Im JSON-Format kann die Eingabe eines Parameters so ausschauen:

# {"KUNDEIN": "001608"}

Bei der JSON-Eingabevariante ist zu beachten, dass der Wrapper-Strukturname nicht in den JSON-Daten enthalten ist.

Ergänzung 42/2017



**REST-Webservices** 

5.5.6 Seite 6 Die Kundenumsatzabfrage beinhaltet die Kundennummer und eine Struktur mit zwei Unterfeldern: FLD1 = Kundenbereich, FLD2 = Vertriebsbereich.

```
{
    "LESKUNDEInput": "0001608",
    "STRUKT1":
    {
        "FLD1": "001",
        "FLD2": "55"
    }
}
```

Mit den HTML-Standards kann die Ausführung eines Webservices und das Füllen der Parameter hinsichtlich potenzieller Fehler überwacht und entsprechend reagiert werden. Dabei liefert der REST-Prozess folgende HTTP-Fehlercodes:

- 200 OK,
- 201 erstellt,
- 202 akzeptiert,
- 204 kein Inhalt,
- 301 permanent entfernt,
- 303 weitere Infos,
- 304 nicht angepasst,
- 307 temporär weitergeleitet,
- 400 fehlerhafte Anforderung,
- 403 verboten,
- 404 nicht gefunden,
- 406 nicht akzeptierfähig,
- 410 verloren,
- 412 Vorprüfung fehlerhaft,
- 415 nicht unterstützter Mediatyp,
- 500 interner Serverfehler,
- 503 Service nicht verfügbar.

Wenn Sie bereits mit den SOAP-Webservices gearbeitet haben, dann kennen Sie die Verwaltungsfunktion der Webservices auf dem integrierten Webservice-Server. Diese ist auch für die REST-Webservices bindend.





## 5.5.6.1 Verwalten von REST-Webservices

IBM hat den IBM-i-Betreibern die Verwaltung der Webservices dahingehend erleichtert, dass mit dem System ein integrierter Webservice-Server geliefert wird. Dieser erlaubt die Definition verschiedener Serverinstanzen, die für sich eigenständig verwaltet werden können. Auf jeder dieser Instanzen können so beliebig viele Webanwendungen konfiguriert und betrieben werden.

Während es in den letzten Jahren eine Beschränkung auf den Einsatz von SOAP-Webanwendungen gab, hat IBM diesen Bereich seit 7.1 mit einem speziellen PTF-Stand mit der Unterstützung für REST-Webanwendungen erweitert.

Hier finden Sie die Voraussetzungen, damit Sie mit Ihrem System i die REST-Webservices nutzen können:

REST-Webservices sind mit dem Betriebssystem IBM i ab der Version 7.1 (PTF-Stand SF99368 Mindeststufe 32) oder der Version 7.2 (PTF-Stand SF99713 Mindeststufe 6) verfügbar.

IBM liefert für die ersten Schritte eine kleine und einfache Webanwendung mit dem System aus. Diese dient der Umrechnung von Fahrenheit in Celsius.

Diese Webanwendung soll beispielhaft für die Implementierung als Webservice im REST-Umfeld genutzt werden.

Sie finden den RPG-Quellcode für das Programm CONVERTTEMP im IFS Ihrer Maschine in folgendem Pfad:

/QIBM/ProdData/OS/WebServices/samples/server/ConvertTemp/ CNVRTTMP.RPGLE

apite Web-Programmierung **REST-Webservices** 5.5.6 📲 Ferne Systeme 🛛 ₽<mark>5.</mark> Team Seite 8  $\bigtriangledown$ 🔓 💫 🗘 🗘 👰 F **E** b in HTTPA D HTTPSVR ICSS imo InfoprintServer b i Java400 b LicenseDoc D OpNavPlugin

a 🗁 OS

Java
 MRI2924
 MRI2929
 MRI2932



ApplicationServer

Pfad für das Webservice-Beispiel ConvertTemp

Dieses Beispiel verfügt über zwei Parameter:

- TempIn (Fahrenheit) und
- TempOut (Celsius).



**REST-Webservices** 



Seite 9

Nachfolgende Abbildung zeigt den Quellcode zum Programm.



## Quellcode von CNVRTTMP

Wenn der REST-Webservice deployt werden soll, dann muss ein integrierter Webservice-Server auf dem System i dafür zur Verfügung stehen. Dieser ist zusammen mit dem Lizenzprogramm 5770-DG1 auf dem System implementiert. Sollten Sie noch keinen Webservice-Server erstellt haben, dann können Sie dies mit den nachfolgenden Teilschritten ausführen.

#### Anmerkung:

Wenn Sie auf Ihrem System bereits einen Webservice-Server konfiguriert haben, dann können Sie diesen für das Testbeispiel nutzen.



# 5.5.6 Seite 10

#### 5.5.6.2 REST-Webservice definieren

Jetzt gilt es, den gewünschten Webservice auf dem Webservice-Server bereitzustellen. Der Einfachheit halber soll an dieser Stelle der von IBM mitgelieferte Webservice CONVERTTEMP für die Umrechnung von Fahrenheit in Celsius genutzt werden. Dieser Webservice verfügt über zwei Parameter:

- Eingabeparameter: Wert in Fahrenheit,
- Ausgabeparameter: Wert für Celsius.

Im Webservice-Server finden Sie im linken Navigationsbereich den Eintrag "Neuen Service implementieren". Diesen selektieren Sie für die Neuanlage eines Webservices.

Voraussetzung für die Implementierung ist, dass das Programm auf dem System i als Programm oder Serviceprogramm definiert wurde und dass die PCML-Informationen vorliegen.

Das von IBM gelieferte Beispiel finden Sie in der Bibliothek QSYSDIR. Es trägt den Namen QIWSSAMPLE.



Neuen Service implementieren

Beginnen Sie also mit der Implementierung des REST-Webservices und klicken auf die Schaltfläche "Neuen Service implementieren".



Web-Programmierung	bite
REST-Webservices	
In der neuen Serverversion müssen Sie in der Folgeanzeige die Art des Services	5.5.6
selektieren. Zur Verfügung stehen hier	Seite 11
SOAP-Services und	

• REST-Services.

Hier wird der REST-Service ausgewählt.



Auswahl Webservice-Typ

Nach der Auswahl bestätigen Sie mit einem Klick auf die Schaltfläche "Weiter".

Kapitel



**REST-Webservices** 

5.5.6 Seite 12 Nun ist anzugeben, wo sich das RPG-Programm befindet. Im Beispiel ist das das Programm QIWSSAMPLE in der Bibliothek QSYSDIR.

Or Market State	A REAL PROPERTY.	
File Edit View Favorites Tools Help       Image: Control of the provided of the provid	http://10.1.2.125:2001/	HTTP/ 🔎 🗧 🖸 🖉 10.1.2.125 - IBM Navigator for i 🖉 HTTP-Server-Administratio ×
BM Web Administration for 1       Image: Netled   2gebring total	File Edit View Favorites Too	ls Help
Notice       Preventure	IBM Web Administration for i	WebSphere
Alle Server   HTTP-Server Application Server Installationen         • Gestartt       • Wei Server /	Konfiguration Verwaltung E	rweitert   Zugehörige Links
● Gestanti       ● WSERVICE - V2.6 (Web-Services) ● <ul> <li>● Web Service Server estation</li> <li>● Web Services Server estation</li> <li>● Web Service argumentize in plementizeria</li> <li>● Web Service argumentizeria</li> <li>● Server-Signerschaften</li> <li>● Server-Fracing</li> <li>● Himplementizeria Bervices involumentizeria</li> <li>● Server-Fracing</li> <li>● Server-Fracing</li> <li>● Frotikolita Berzegien</li> <li>● Protokolita Berzegien</li> <li>● Services</li> <li>● Implementizeria Berzigen</li> <li>● Web Server-Fracing</li> <li>● Protokolita Berzegien</li> <li>● Services</li> <li>● Den Namen der IBM +Bibliothek und des ILE-Programmobjekts selbst. Dies ist der schneilste und auch der emptohlene Weg zu Leoblethame:</li> <li>● Den Namen der IBM +Bibliothek.</li> <li>● Protokolita Berzegien</li> <li>● Services</li> <li>● Protokolita Berzegien</li> <li>● Services</li> <li>● Den Valence IBM +Bibliothek.</li> <li>■ Dieleberstimmung</li> <li>● Protokolita Berzegien</li> <li>● Services</li> <li>● Dieleberzimmung</li> <li>● Prot</li></ul>	Alle Server   HTTP-Server Appl	lication Server Installationen
<ul> <li>Alignmeine Tasi's und Assistenten Web Service Server erstellen Anvendnungserver erstellen Meb Sarvice Assistenten Meb Rever Tracing Meth Dep Monitor Meb Log Monitor Me</li></ul>	🧕 Gestartet 🚺 🔲 🧭 Server:	WSERVICE - V2.6 (Web-Services) V
<ul> <li>Aniwendungsserver aristellen WebSphere Portal erstellen</li> <li>Web-Service-Assistenten Die Neuen Service implementieren Die Sales SL</li> <li>Server-Eigenschaften Di Server-Traing Die HTTP-Server anzeigen</li> <li>Services Di Implementierte Services verwatter</li> <li>Petherbostimung Die Protokolle anzeigen Web Log Monitor Web Log Monitor</li> <li>Web Log Monitor Web Log Monitor</li> <li>Weiter Abbrechen</li> </ul>	Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTP-Server erstellen	WSERVICE > Implementierte Services verwalten > Neuen Service implementieren
Web-Service-Assistenten         Web-Service-Assistenten         Stevice-Assistenten         Stontgureen         Disabel SSL         Service-Eigenschaften         Disaver-Fragenschaften         District-Polyticthen         Derotokolle narzeigen         Methodshaften         District-Dipolekthyn: <td< td=""><td>Anwendungsserver erstellen</td><td>Neuen Service implementieren</td></td<>	Anwendungsserver erstellen	Neuen Service implementieren
Web-Service-Assistenten     Web-Service implementation     Service-Tracing     Betrive-Services versuate     Service-Tracing     S	g websphere i onal erstellen	Speicherposition von IBM i-Programmobjekt angeben - Schritt 2 von 9
<ul> <li>Server-Eigenschaften Bigenschaften Berverzahren</li></ul>	Web-Service-Assistenten Neuen Service implementieren SSL konfigurieren Disable SSL	Das auszulagernde &OS-Programmobjekt muss ein auf dem System vorhandenes ILE- oder Serviceprogrammobjekt (*PGM bzw. *SRVPGM) sein.
<ul> <li>b HTTP-Second b HTP-Second b H</li></ul>	Server-Eigenschaften     Eigenschaften     Server-Tracing	Das Programmobjekt für den Web-Service angeben. 🤨
<ul> <li>Services         <ul> <li>Bit Speicherposition des Programmobjekts können Sie angeben, indem Sie sowohl den Namen der Bibliothek eingeben, die das Programmobjekt index inden Namen des Programmobjekts stelbst. Dies ist der schnellste und auch der empfohlene Weg zu Lokalisierung des Programmobjekts.</li> <li>Bibliotheksname: OSYSDIR</li></ul></li></ul>	B HTTP-Server anzeigen	Den Namen der IBM i-Bibliothek und des ILE-Programmobjekts angeben (Empfohlen)
► Fehterbestimmung b Protokolle anzeigen d Web Log Montor d <sup>2</sup> Ubersicht anzeigen ULE-Objekttyp: ● -SRVPGM ○ *PGM ○ Das integrierte Dateisystem nach dem IBM I-Programmobjekt durchsuchen Zurück Weiter Abbrechen	✓ Services	Die Speicherposition des Programmobjekts können Sie angeben, indem Sie sowohl den Namen der Bibliothek eingeben, die das Programmobjekt enthält, als auch den Namen des Programmobjekts selbst. Dies ist der schneitste und auch der empfohlene Weg zu Lokalisierung des Programmobjekts.
LE-Objektname: QIWSSAMPLE Ubersicht anzeigen LE-Objekttyp: ●•SRVPGM ○•PGM ○ Das integrierte Dateisystem nach dem IBM i-Programmobjekt durchsuchen Zurück Weiter Abbrechen	▼ Fehlerbestimmung	Bibliotheksname: QSYSDIR
u <sup>p</sup> <sup>a</sup> Ubersicht anzeigen ILE-Objekttyp: ●*SRVPGM ○*PGM ○ Das integrierte Dateisystem nach dem IBM i-Programmobjekt durchsuchen Zurück Weiter Abbrechen	4 Web Log Monitor	ILE-Objektname: QIWSSAMPLE
Das integrierte Dateisystem nach dem IBM i-Programmobjekt durchsuchen      Zurück Weiter Abbrechen	് Ubersicht anzeigen	ILE-Objekttyp:   • *SRVPGM · *PGM
Zurück Weiter Abbrechen		O Das integrierte Dateisystem nach dem IBM i-Programmobjekt durchsuchen
		7.000 (0.000) (0.000)
		Zuruck weiter Abbrechen

Aufrufangaben IBM-i-Programm

Bestätigen Sie auch hier mit einem Klick auf "Weiter".



Jetzt kommt eine Neuerung im Vergleich zum SOAP-Service, denn nun müssen die Details zu den HTTP-Anforderungen hinterlegt werden. - 0 **X** 🗲 🕣 🥖 http://10.1.2.125:2001/HTTP/ 🔎 – 🗟 🖉 💋 🦪 10.1.2.125 - IBM Navigator for i 🛛 🧔 HTTP-Server-Administratio.... 🗙 ት 🖈 Edit View Favorites Tools Help IBM Web Administration for i Konfiguration Verwaltung Erweitert | Zugehörige Links (?) WebSphere. IBM Alle Server | HTTP-Server Application Server Installationen Gestartet D D Server: WSERVICE - V2.6 (Web-Services) 
 Allgemeine Tasks und Assistenten
 WSERVICE > Implementierte Services verwalten > Neuen Service implementieren

 Weh Services Server erstellen
 JAmmendungsserver erstellen

 JAmmendungsserver erstellen
 Neuen Service implementieren
 Neuen Service implementieren WebSphere Portal erstellen Namen für Service angeben - Schritt 3 von 9 ✓ Web-Service-Assistenten Neuen Service implementieren SSL konfigurieren Disable SSL Der auszulagernde Web-Service ist eine Ressource. Die URI-Pfadvorlage identifiziert übereinstimmende Muster für ankommende HTTP-Anforderungen. Der Pfad ist relativ zur Kontextroot und kann eine einfache Zeichenfolge sein oder aus einem oder mehreren \_ Vorlagenparametern bestehen, die reguläre Ausdrücke enthalten können, um den zulässigen Umfang weiter einzuschränken. Server-Eigenschaften
 Eigenschaften
 Server-Tracing
 HTTP-Server anzeigen Ressourcenname: Temperatur Servicebeschreibung: Temperatur Umrechnung URI-Pfadvorlage: /{temp: \d+} e.g. /temperature, /temperature/{temp:\d+} · Services b Implementierte Services verwalten Fehlerbestimmung
 Protokolle anzeit Protokolle anzeigen Web Log Monitor Übersicht anzeigen Zurück Weiter Abbrechen 

Servicenamen festlegen

Geben Sie hier als Ressourcenname einen eindeutigen Namen an. Das Feld "Servicebeschreibung" kann zur Angabe einer Beschreibung genutzt werden.

Im Feld "URI-Pfadvorlage" dient der URI Angabe für die Verarbeitung der Anforderungen - hier wird hinterlegt, wie Ressourcen und Ressourcenmethoden zusammenarbeiten.

Ein Parameter wird in eckigen Klammern definiert. Achten Sie darauf, dass der Parameter mit einem Slash beginnt. Im Beispiel ist das /{temp}. Allerdings gibt es noch eine Erweiterung in Form von \d+. Das bedeutet, dass als Parameter nur Ziffern angegeben werden können (D = digits).



Seite 13

Web-Programmierung
RFST-Webservices





**REST-Webservices** 

5.5.6 Seite 14 Mit einem Klick auf "Weiter" sind nun die auf Basis der PCML-Definitionen ausgelesenen Parameter des zu implementierenden Webservices zu finden.

A R R D AVEN		real manager instruct in				x
← → Mttp://10.1.2.125:2001/0	HTTP4 🔎 🔻 🗟 🖒	🥖 10.1.2.125 - IBM Navigator for i	HTTP-Server-	Administratio	. × గి ర	7 🔅
File Edit View Favorites Tool	s Help					
IBM Web Administration for i Konfiguration Verwaltung Et	weitert   Zugehöri	ge Links			(?) WebSphere.	M
Alle Server   HTTP-Server Appl	ication Server	Installationen				
🕒 Gestartet 🚺 🔲 🧭 Server: 🛽	WSERVICE - V2	6 (Web-Services) 🗸				
Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTP-Server erstellen	WSERVICE > Imp	olementierte Services verwalten > Neue	n Service impleme	entieren		^
Anwendungsserver erstellen	Neuen Servi	ice implementieren				
o websphere Portal erstellen	Exportprozedure	en zur Auslagerung als Web-Service	auswählen - Sc	hritt 4 von 9		
<ul> <li>▼ Web-Service-Assistenten</li> <li>Neuen Service implementieren</li> <li>☆ SSL konfigurieren</li> <li>☆ Disable SSL</li> </ul>	Exportierte Proze besteht aus einer anschließend zur eine einzige Proz	duren sind Eingangspunkte in ein Progr Gruppe eigenständiger Anweisungen in aufrufenden Programm zurückkehrt. E edur.	rammobjekt, die W n einer höheren Pr Ein Serviceprogran	/eb-Service-Op rogrammierspra nm enthält min	perationen zugeordnet werden. Eine Prozedur ache, die eine bestimmte Task ausführt und destens eine Prozedur. Ein Programm enthält nur	
Server-Eigenschaften     D Eigenschaften     D Server-Tracing     D HTTP-Server anzeigen     Services	Die nachfolgende ausgelagert werd Parameterattribut wird. Für Feldgru	Tabelle enthält alle exportierten Prozec en kann. Blenden Sie die Prozedurzeile 'Verwendung' hat Einfluss darauf, welc ppentypparameter kann eine Änderung	duren, die in dem l ein, um die Stand he Daten von Clie des Felds 'Anzahl	Programmobje lardeinstellung nts gesendet w ' zu einem bes:	kt enthalten sind, das durch diesen Web-Service en für die Prozedurparameter zu ändem. Das verden und was vom Web-Service zurückgegeben seren Leistungsverhalten des Web-Services führer	n.
B implementiente Services verwalten	Detect length field	ds 🖌				
▼ Fehlerbestimmung	Exportprozedure	n: 😢				
4 Web Log Monitor	Auswählen	Prozedurname/Parametername	Verwendung	Datentyp		
Jor Obersicht anzeigen		▼ CONVERTTEMP				
		TEMPIN	Eingabe 🗸	char		
		TEMPOUT	Ausgabe 🗸	char		$\sim$
	Zurück Weite	Abbrechen			<u></u>	

#### Parameterdefinitionen

Die Parameter lassen sich nicht in Bezug auf deren Namen oder Art ändern. Lediglich die Art der Verwendung ist hier anpassbar, also die Angabe, ob es sich beim Parameter um einen Eingabeparameter, einen Ausgabeparameter oder um beides handelt.







← → Attp://10.1.2.125:2001/	HTTP/ 🔎 👻 🖒 🥥 10.1.2.125 - IBM Navigator for i 🛛 HTTP-Server-Administratio ×					
File Edit View Favorites Too	s Help					
IBM Web Administration for i	WabSabara					
Konfiguration Verwaltung E	weitert   Zugehörige Links					
Alle Server   HTTP-Server Appl	ication Server Installationen					
🕒 Gestartet 🕞 🔲 🧭 Server:	WSERVICE - V2.6 (Web-Services) 🗸					
<ul> <li>Allgemeine Tasks und Assistenten</li> </ul>	Informationen zur Ressourcenmethode angeben - Schritt 5 von 9					
HTTP-Server erstellen	Prozeduren werden Ressourcenmethoden zugeordnet. Jede Ressourcemethode muss für die Verarbeitung von Clientanforderungen definiert					
WebSphere Portal erstellen	werden, indem eine HTTP-Anforderungsmethode einer Ressourcenmethode zugeordnet wird.					
Web-Service-Assistenten	Informationen zur Ressourcenmethode angeben 🥹					
SSL konfigurieren						
Disable SSL	Procedure name: CONVERTTEMP					
<ul> <li>Server-Eigenschaften</li> <li>Eigenschaften</li> </ul>	HTTP-Anforderundsmethode:					
B Server-Tracing	URI-Pfadvorlage für die Methode: *NONE oder V					
	HTTP response code output parameter: *NONE V					
<ul> <li>Services</li> <li>Implementierte Services verwalten</li> </ul>	HTTP header array output parameter: *NONE V					
- Fehlerhestimmung	Zulässige Eingabemedientypen: *ALL oder V					
Protokolle anzeigen Web Log Monitor	Zurückgegebene Ausgabemedientypen: *XML_AND_JSON oder					
P Übersicht anzeigen	Whether to wrap input parameters:					
	O Wrap input parameters					
	Do not wrap input parameters					
	Eingabeparameterzuordnungen:					
	TEMPIN char - Auswählen - V *NONE oder V					
		~				
	<					
	Zurück Weiter Abbrechen					
		4				
5		2				

Ressourcenmethoden festlegen

Beim ersten Parameter wird die HHTP-Anforderungsmethode angegeben. Dabei kann zwischen den folgenden Methoden gewählt werden:

- GET Lesen
- PUT Ändern bestehender Elemente
- POST Neuanlage von Elementen
- DELETE Löschen von Elementen

Damit der Webservice mit möglichst wenigen manuellen Eingriffen und Konfigurationen automatisch erstellt werden kann, bedarf es einiger weniger Anpassungen im RPG.



5.5.6 Seite 16 Zunächst müssen der Nicht-RPG-Anwendung Informationen übergeben werden, mittels derer das RPG-Programm korrekt aufgerufen werden kann. Diese Informationen beinhalten zum Beispiel:

- die Bibliothek, in der das aufzurufende Programm existiert,
- den Namen des aufzurufenden Programms,
- die Definition der Parameter getrennt nach Eingabe, Ausgabe, Ein-/Ausgabe.

Diese Informationen werden dem Webservice auf Basis einer PCML-Definition bereitgestellt. Diese lässt sich automatisch generieren, indem die Header-Anweisung h jaNOMAIN PGMINFO(\*PCML:\*MODULE) im RPGLE-Programmcode eingefügt wird.

Beim Einsatz eines REST-Webservices sollen sich vor dessen Erstellung einige Gedanken hinsichtlich der HTTP-Methoden gemacht werden, die im REST-Webservice verarbeitet werden sollen.

Die "URI-Pfadvorlage für die Methode" dient dem Zuordnen der Anforderungen zu den Anforderungsmethoden. Im Beispiel bleibt dieser Parameter auf dem Standardwert \*NONE.

In folgender Anzeige ist die Bedeutung der unterschiedlichen Parameter ersichtlich.

Procedure Name:

Name der Prozedur. Dieser wurde durch die bisherigen Eingaben bereits voreingestellt.

- URI Path Template for Resource: Hier ist die URI-Pfadschablone für die Ressource anzugeben.
- HTTP-Anforderungsmethode:

Hier muss die HTTP-Anforderungsmethode angegeben werden. Als Standardwert für diesen Parameter ist "GET" voreingestellt. Über das Selektionssymbol kann zudem zwischen den folgenden Anforderungsmethoden gewählt werden:

- GET,
- PUT,
- DELETE und
- POST.

#### • URI-Pfadvorlage für die Methode:

Die URI-Pfadvorlage für die Methode dient dem Zuordnen von Anforderungen zu den Ressourcenmethoden.





## • HTTP Response Code Output Parameter:

Hier kann ein Ausgabeparameter vorgegeben werden, der den HTTP-Antwortcode aufnimmt. Damit kann auf unterschiedliche HTTP-Ergebnisse reagiert werden. Der Parameter ist zwingend als Integerwert anzugeben.

## • HTTP Header Array Output Parameter:

Hier kann optional ein Ausgabeparameter für die Aufnahme von HTTP-Headern, die an den Client übertragen werden, definiert werden. Wird dieser Parameter gewünscht, dann muss dieser als Array (Char) angelegt werden.

## • Zulässige Eingabemedien:

Hier können die zulässigen MIME-Typen hinterlegt werden, die mit der Ressourcenmethode verarbeitet werden dürfen.

Entweder übernehmen Sie den Standardwert \*ALL oder selektieren im Auswahlfenster die gewünschten Eingabemedien. Gültige Werte sind:

- \*ALL,
- \*XML,
- \*JSON und
- \*XML\_and\_JSON.

## Zurückgegebene Ausgabemedientypen:

Hier sind die MIME-Typen zu hinterlegen, die bei der Ressourcenmethode zurückgegeben werden sollen.

#### • Whether to Wrap Input Parameters:

Dieser Wert definiert, ob die Eingabeparameter in einer Struktur übergeben werden.



REST-Webservices

# 5.5.6 Seite 18

Kapitel

← (⇒)	HTTP/ 🔎 🗧 😋 10.1.2.125 - IBM Navigator for i 🖉 HTTP-Server-Administratio × 👘 🛣				
File Edit View Favorites Too	ls Help				
IBM Web Administration for i	WebSphere				
Konfiguration Verwaltung E	weitert   Zugehörige Links				
Alle Server   HTTP-Server Appl	ication Server Installationen				
🕒 🕒 Gestartet 🚺 🔲 💋 Server:	WSERVICE - V2.6 (Web-Services) V				
Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTP-Server erstellen     Anwendungsserver erstellen	Neuen service implementieren Informationen zur Ressourcenmethode angeben - Schritt 5 von 9				
WebSphere Portal erstellen	werden, indem eine HTTP-Anforderungsmethode einer Ressourcenmethode zugeordnet wird.				
<ul> <li>▼ Web-Service-Assistenten</li> <li>☆ Neuen Service implementieren</li> <li>☆ SSL konfigurieren</li> <li>☆ Disable SSL</li> </ul>	Informationen zur Ressourcenmethode angeben 3				
- Server-Figenschaften	Procedure name: CONVERTTEMP				
B Eigenschaften	URI path template for resource: //temp:/d+}				
B Server-Tracing B HTTP-Server anzeigen	HTTP-Antorderungsmeunode. GET V				
- Satricas					
Implementierte Services verwalten					
▼ Fehlerbestimmung	Tulässiae Finashamadianhusan:				
Protokolle anzeigen Web Log Monitor	Zulassige Eingabene dienhinen: ALL Oder V				
P Übersicht anzeigen	Zuruckgegebene Ausgabemedientypen: <sup>*</sup> JSON oder V				
	Do not wrap input parameters				
	Eingabeparameterzuordnungen:				
	Parametername Datentyp Input source Kennung Default Value				
	TEMPIN char PATH_PARAM V temp V NONE oder V				
	Zurück Weiter Abbrechen				

Input-Ressource \*PATH\_PARAM

Nach der Eingabe der erforderlichen Parameter klicken Sie erneut auf "Weiter".



In der nächsten Anzeige muss der Benutzer angegeben werden, unter dem der Service auf dem System i ausgeführt werden soll. Dabei kann wahlweise die Benutzer-ID verwendet werden, mit der der Server ausgeführt wird, oder es kann ein existierender IBM-i-Benutzer angegeben werden. Diese Angabe wirkt sich unter anderem auf die Objektberechtigungsprüfung aus. Deshalb bist es sinnvoll und wichtig, einen Benutzer anzugeben, der auf dem System i über die notwendigen Berechtigungen für die Programmausführung verfügt.



Benutzer für Webservice-Ausführung

Es folgt wieder ein Klick auf die Schaltfläche "Weiter".



Ein ITP Handbuch



**REST-Webservices** 

5.5.6 Seite 20 Analog der Bibliotheksliste in einem Job muss nun angegeben werden, welche Bibliotheken in welcher Reihenfolge für die erfolgreiche Ausführung des Jobs benötigt werden. Diese Bibliotheken sind in der nachfolgenden Anzeige zu definieren.

← → Attp://10.1.2.125:2001/	HTTP: 🔎 - 🗟 C 🧭 10.1.2.125 - IBM Navigator for i 🖉 HTTP-Server-Administratio ×
File Edit View Favorites Too	ls Help
IBM Web Administration for i	WebSphere
Konfiguration Verwaltung E	weitert   Zugehörige Links
Alle Server   HTTP-Server Appl	ication Server Installationen
🧕 Gestartet 🚺 🔲 💋 Server:	WSERVICE - V2.6 (Web-Services) V
Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTP-Server erstellen	WSERVICE > Implementierte Services verwalten > Neuen Service implementieren
Anwendungsserver erstellen	Neuen Service implementieren
Web Orning Assistants	Bibliotheksliste angeben - Schritt 7 von 9
Web-Service-Assistenten           Deuen Service implementieren           SSL konfigurieren           Disable SSL	Die Funktionalität des IBM i-Programms, das als Web-Service ausgelagert werden soll, hängt möglicherweise von anderen IBM i-Programme im System ab. Geben Sie alle Bibliotheken an, die Programme enthalten, von denen die Web-Service-Programme abhängig sind. Wenn keine Bibliotheksangabe erfolgt, wird die Standardbibliothek verwendet.
<ul> <li>Server-Eigenschaften</li> </ul>	Die Position der Bibliotheksliste für diesen Web-Service angeben:
B Server-Tracing	O Bibliotheken vor dem Abschnitt mit den Benutzerbibliotheken in die Bibliotheksliste einfügen
B HTTP-Server anzeigen	Bibliotheken hinter dem Abschnitt mit den Benutzerbibliotheken in die Bibliotheksliste einfügen
▼ Services	Bibliothekslisteneinträge: 3
Inplementerte Services verwaiten	Bibliotheksname
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> </ul>	O QSYSDIR
4 Web Log Monitor	Hinzufügen Alles entfernen
a obersicht anzeigen	
	Zurück Weiter Abbrechen

Bibliotheksliste festlegen

IBM stellt an dieser Stelle bereits die Bibliothek ein, in der sich das basierende Programm oder Serviceprogramm befindet, das für den Webservice Pate steht. Mit einem Klick auf die Schaltfläche "Hinzufügen" können anschließend weitere Bibliotheken angegeben werden. Dabei muss darauf geachtet werden, dass die Bibliotheksliste sowohl in Bezug auf die Bibliotheken als auch in Bezug auf die Reihenfolge genau festzulegen ist.

Es folgt ein Klick auf die Schaltfläche "Weiter".



Web-Programmierung REST-Webservices

Kapitel

5.5.6

Seite 21

(a) (a) (b) (c) (c) (c) (c) (c) (c) (c) (c) (c) (c	HTTP/ D - 🗟 C 🥥 10.1.2.125 - IBM Navigator for i 🖉 HTTP-Server-Administratio ×
File Edit View Favorites Too	ls Help
IBM Web Administration for i	(2) WebSphere
Konfiguration Verwaltung E	weitert   Zugehörige Links
Alle Server   HTTP-Server Appl	ication Server Installationen
Gestartet D 2 Server:	WSERVICE - V2.6 (Web-Services) V
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> </ul>	Neuen Service implementieren
HTTP-Server erstellen	Zu übergebende Transportinformationen angeben - Schritt 8 von 9
WebSphere Portal erstellen	Transportinformationen angeben, die an den Implementierungscode des Web-Service übergeben werden sollen.
▼ Web-Service-Assistenten	Informationan, die an den Implementierungssonde des Web Service übergeben werden 2
Neuen Service implementieren SSL konfigurieren	informationen, die an den implementerungscode des web-service übergeben werden 🗸
Disable SSL	Transportmetadaten angeben:
<ul> <li>Server-Eigenschaften</li> <li>Eigenschaften</li> </ul>	Transportmetadaten
Server-Tracing     NUTTE Server anzeigen	
G HTTF-Server anzeigen	
<ul> <li>Services</li> <li>Implementierte Services verwalten</li> </ul>	
 Esklashastimana	
Protokolle anzeigen	
🚽 Web Log Monitor 🖑 Übersicht anzeigen	SERVER_NAME
	SERVER_PORT
	HTTP-Header angeben:
	HTTP-Header
	Es sind keine Einträge für diese Tabelle vornanden.
	Zurück Weiter Abbrechen

Details Transportmetadaten



Kapitel

# Web-Programmierung

**REST-Webservices** 

## 5.5.6 Seite 22

		T		×
A ttp://10.1.2.125:2001/	HTTP/ 🔎 🔻 🗟 🖒 <i>(</i> 10.1.2	.125 - IBM Navigator for i 🖉 HTTP-Server-Administratio 🗙	🔐 🖒	ŝ
File Edit View Favorites Too	ls Help			
IBM Web Administration for i			WebSphere	
Konfiguration Verwaltung E	rweitert   Zugehörige Links			÷
Alle Server   HTTP-Server Appl	ication Server Installation	ien		
🕒 Gestartet 🕞 🔲 💋 Server:	WSERVICE - V2.6 (Web-Se	rvices) 🗸		
✓ Allgemeine Tasks und Assistenten	WSERVICE > Implementierte	Services verwalten > Neuen Service implementieren		
HTTP-Server erstellen     Anwendungsserver erstellen	Neuen Service impl	ementieren		
WebSphere Portal erstellen	Zusammenfassung - Schrit	t 9 von 9		
✓ Web-Service-Assistenten	Zubunnenhubbung bunn	13 10/1 3		
<ul> <li>한 Neuen Service implementieren</li> <li>한 SSL konfigurieren</li> <li>한 Disable SSL</li> </ul>	Wenn Sie auf Fertig stellen	klicken, wird der Web-Service implementiert.		
<ul> <li>Server-Figenschaften</li> </ul>	Service Methods Anfor	derungsinformationen		
B Eigenschaften				
B HTTP-Server anzeigen	Ressourcenname:	Temperartur Temperatur Umrecheung		
- Capricas	Serviceinstallationspfad:	/www/wservice/webservices/services/Temperartur		
<ul> <li>Services</li> <li>Implementierte Services verwalten</li> </ul>	URI-Pfadvorlage:	/{temp:\d+}		
	Benutzer-ID für Service:	ZEIG		
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> </ul>	Programm: Pibliothokelisto für Sorvico	/QSYS.LIB/QSYSDIR.LIB/QIWSSAMPLE.SRVPGM		
4 Web Log Monitor	Dibilotrieksliste für Service	USTSDIK		
is obsidiate an Eargon				
	Zurück Fertig stellen	Abbrechen		
			6	

Zusammenfassung

Damit sind die für den REST-Webservice erforderlichen Angaben komplettiert. Mit einem Klick auf die Schaltfläche "Fertig stellen" wird der Webservice auf dem Server angelegt.



Web-	Progra	mmieru	1 <b>g</b>

apitel

5.5.6

Seite 23

Zeitgleich erscheint die Implementierungsanzeige mit dem Status "Wird installiert".



Webservice-Verwaltungssicht

Ergänzung 42/2017

Kapitel

# Web-Programmierung

**REST-Webservices** 

# \* ..... 5.5.6 (C) (D) ( Seite 24 (?) WebSphere. IBM Alle Server | HTTP-Server Application Server Installationen Gestartet D D Server: WSERVICE - V2.6 (Web-Services) V Allgemeine Tasks und Assistenten Web Services Server erstellen Thr-Server erstellen WebSphere Portal erstellen WebSphere Portal erstellen Stand vom 08.07.2016 14:09:54. • Web-Service-Assistenten \* Neuen Service implementieren \* SSL konfigurieren Disable SSL Op SSL konfigurieren Impertienulerte Services: Imperienulerte Services: Imperienulerte Service Service Status Type Stattyp Service definition • Service-Eigenschaften Implementieren Stoppen Eigenschaften Service testen D Servicer anzeigen Stoppen Eigenschaften Deinstallieren Aktualisieren Service testen Services Implementierte Services verwalten Fehlerbestimmung Protokolle anzeigen Web Log Monitor Übersicht anzeigen Schließen

#### Gestarteter Webservice

Jetzt muss nur noch sichergestellt werden, dass der neu implementierte Webservice auch gestartet ist. Das ist am Status (gestartet) und dem grünen Symbol erkennbar.



## 5.5.6.3 **REST-Webservice testen**

Abschließend heißt es, den Service testen. Dazu klicken Sie auf die Schaltfläche "Service testen", die Sie bereits von den SOAP-Webservices her kennen. Doch die Schaltfläche ist deaktiviert.

Das hat nichts mit Ihrer Vorgehensweise zu tun, sondern ist leider eine Tatsache. Ein SOAP-Webservice auf System i kann mit dem integrierten Test-Client auf seine Funktion hin geprüft werden, doch besteht genau diese Option im Zusammenhang mit den REST-Webservices nicht. Doch soll hier die Stärke der REST-Webservices genutzt werden. Und diese lautet: Der REST-Webservice verwendet schlicht eine HTTP-GET-Methode, die mit jedem gängigen Browser simuliert bzw. genutzt werden kann.

Also öffnen Sie ein neues Browser-Fenster und geben dort die URL für Ihren neuen Webservice an. Diese können Sie in den Eigenschaften des neuen Webservices schnell finden. Schauen Sie sich dazu folgende Abbildung an.



IBM Web Administration for i	invoitort   Zugol	börige Link			
verwaitung	rweiten   Zugei	Tonge Link	5		
Alle Server   HTTP-Server App	lication Serve	er Instal	lationen		
💊 Gestartet 🚺 🔲 💋 Server:	WSERVICE -	V2.6 (We	b-Services) 🗸		
Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTP-Server erstellen     Anwendungsserver erstellen     WebSphere Portal erstellen	WSERVICE >	Implement	tierte Services verw haften	<u>valten</u> > Service-Eiger	nschaften
Web-Service-Assistenten	Allgemein	Methods	Bibliotheksliste	Verbindungspool	Anforderungsinformationen
Neuen Service implementieren     SSL konfigurieren     Disable SSL	Serviceinforn Resource Nat	mationen ( me:	Personal Temperartu	ır	
✓ Server-Eigenschaften	Resource des URI path tem Starttyp :	cription: plate:	Temperatu /{temp:\d+}	r Umrechnung	
B HTTP-Server anzeigen	Serviceinstall	ationspfad:	/www/WSE	RVICE/webservices/s	/services/Temperartur
<ul> <li>Services</li> <li>Implementierte Services verwalten</li> </ul>	Base resource	e URL:	http://10.1.2	2.125:10010/web/serv	vices/Temperartur
<ul> <li>▼ Fehlerbestimmung</li> <li>▶ Protokolle anzeigen</li> <li>↓ Web Log Monitor</li> <li>⊲<sup>b</sup> Übersicht anzeigen</li> </ul>	Benutzer-18-1	är diesen C	Dervice. ZE18	nutzer-ID des Servers	s aktualisieren, so dass sie die Berechtigung *USE für diese Benutzer-ID erh

## Aufruf-URL

Der Parameter "Base resource URL" beinhaltet die Aufruf-URL, allerdings ohne den notwendigen Eingabeparameter. Diesen müssen Sie bei der URL-Eingabe mitberücksichtigen bzw. eingeben. Zum Beispiel wäre dies für die Ermittlung des Celsius-Werts für 50 Grad Fahrenheit folgende URL: http://ASZEBIS:10010/web/services/Temperatur/50





**REST-Webservices** 

5.5.6 Seite 26 Nach dem Bestätigen der Eingabe wird der REST-Webservice auf dem System i aufgerufen. Das Ergebnis sehen Sie jedoch nicht sofort, sondern Sie erhalten einen Hinweis mit der Frage, ob das JSON-Format gespeichert oder geöffnet werden soll.

	http://101.2.125:10010/web/sen $\mathcal{P} \star \mathcal{O}$ 🦉 Blank Page X	th ☆ th
File Edit V	iew Favorites Tools Help	
	Do you want to open or save 50.json (18 bytes) from 10.1.2.125? Open Save ▼ Cancel >	c .

JSON-Ergebnis

Das Ergebnis soll nur geöffnet werden, und dementsprechend wird die Option "Öffnen" gewählt.



Kapitel

Für die JSON-Ergebnisanzeige können Sie unterschiedliche Tools nutzen. Im Beispiel genügt die Anzeige in Notepad. Deshalb wird Notepad als Anwendung selektiert. Damit erhalten Sie die Anzeige des Ergebnisses des REST-Webservice-Aufrufs. 5.5.6 Seite 27

_		
	🗍 50.json - Notepad	
	File Edit Format View Help	
	{"TEMPOUT":"9.99"}	*
		*
-		

Werte in Notepad

In der Ergebnisanzeige finden Sie den Namen des Parameters und den ermittelten Wert.

Demzufolge entsprechen 50 Grad Fahrenheit 10 Grad Celsius – das IBM hier 9,99 zurückliefert, bietet Anlass für ein eigenes PTF ...

Damit haben Sie einen lauffähigen und universell einsetzbaren REST-Webservice auf der Basis eines RPG-Programms generiert.



5.5.6.4 Seite 28

## 5.5.6.4 REST-Serviceprogramm Kundenstamm

Beispielhaft soll Ihnen aufgezeigt werden, wie ein REST-Webservice auf Basis eines einfachen RPG-Serviceprogramms erstellt werden kann.

Dazu wird eine Tabelle KUNDENPF verwendet, die mit nachfolgender SQL-Anweisung erstellt werden kann.

```
CREATE TABLE zeig/kundenpf
("Kundennr" FOR COLUMN KundenNr CHAR (9) NOT NULL,
"Vorname" FOR COLUMN Vorname CHAR (50) NOT NULL,
"Nachname" FOR COLUMN NachName CHAR (50) NOT NULL,
"Kundentyp" FOR COLUMN KdTyp CHAR (10) NOT NULL,
PRIMARY KEY ( KundenNr ))
RCDFMT KundenR
```

Erstellen einer Kundenstammdatei

Anmerkung:

Natürlich kann die Tabelle bzw. Datei auch mit DDS-Mitteln erzeugt werden.

Jetzt wird die Tabelle noch mit ein wenig Leben gefüllt.

```
insert into zeig/kundenpf (kundenNr, vorname, nachname, kdtyp) value
s ('1000', 'Willi', 'Winzig', 'Endkunde'), ('1001', 'Max', 'Schmidt'
, ,Markt'), (,1002', ,Leo', ,Leopard', ,Endkunde')
```

Der REST-Webservice soll flexibel eingesetzt werden können. Dazu werden im Serviceprogramm folgende Prozeduren definiert:

• Create

Neuanlage eines Kundenstammsatzes

- HTTP-Methode: POST
- URI: /contect-root/kunde
- Update

Ändern eines bestehenden Kundenstammsatzes

- HTTP-Methode: PUT
- URI: /context-root/kunde
- Delete Löschen eines bestehenden Kundenstammsatzes
  - HTTP-Methode: DELETE
  - URI: /context-root/kunde/id





• GetAll

Abrufen aller Kundenstammsätze

- HTTP-Methode: GET
- URI: /context-root/kunden
- GetByNr

Abrufen eines spezifischen Kunden mit Kundennummer

- HTTP-Methode: GET
- URI: /contect-root/kunden/id

Jede der Prozeduren wird mit einer HTTP-Methode verbunden. Wird jetzt das RPG aufgerufen, dann kann unter anderem auch auf verschiedene Ereignisse reagiert werden.

Dabei bedienen Sie sich der standardisierten HTTP-Statusmeldungen. Diese sind unter anderem folgende:

Code	Nachricht	Bedeutung
200	ОК	Die Anfrage wurde erfolgreich bearbeitet, und das Ergebnis der Anfrage wird in der Antwort übertragen.
201 Created Die Anfrage Ressource w "Location"-H erstellten Re		Die Anfrage wurde erfolgreich bearbeitet. Die angeforderte Ressource wurde vor dem Senden der Antwort erstellt. Das "Location"-Header-Feld enthält eventuell die Adresse der erstellten Ressource.
202	Accepted	Die Anfrage wurde akzeptiert, sie wird aber zu einem späteren Zeitpunkt ausgeführt. Das Gelingen der Anfrage kann nicht garantiert werden.
203	Non- Authoritative Information	Der Server agiert als "Transforming Proxy", erhielt eine 200-OK-Antwort von der Quelle und antwortet mit einem veränderten Dokument der Quelle.
204	No Content	Die Anfrage wurde erfolgreich durchgeführt. Die Antwort enthält jedoch bewusst keine Daten.
205	Reset Content	Die Anfrage wurde erfolgreich durchgeführt. Der Client soll das Dokument neu aufbauen und Formulareingaben zurücksetzen.
206	Partial Content	Kann einen Client über Teil-Downloads informieren (wird zum Beispiel Wget genutzt, um den Download-Fortschritt zu überwachen oder einen Download in mehrere Streams aufzuteilen).
207	Multi- Status	Die Antwort enthält ein XML-Dokument, das mehrere Statuscodes zu unabhängig voneinander durchgeführten Operationen enthält.

Erfolgreiche Anweisungen (Code 2xx)



Kapitel

# Web-Programmierung

REST-Webservices

# 5.5.6.4 Seite 30

## Umleitungen (Code 3xx)

Code	Nachricht	Bedeutung
300	Multiple Choices	Die angeforderte Ressource steht in verschiedenen Arten zur Verfügung. Die Antwort enthält eine Liste der verfügba- ren Arten. Das "Location"-Header-Feld enthält eventuell die Adresse der vom Server bevorzugten Repräsentation.
301	Moved Permanently	Die angeforderte Ressource steht ab sofort unter der im "Location"-Header-Feld angegebenen Adresse bereit (auch Redirect genannt). Die alte Adresse ist nicht länger gültig.
302	Found (Moved Temporarily)	Die angeforderte Ressource steht vorübergehend unter der im "Location"-Header-Feld angegebenen Adresse bereit. Die alte Adresse bleibt gültig. Die Browser folgen meist mit einem GET, auch wenn der ursprüngliche Request ein POST war.
303	See Other	Die Antwort auf die durchgeführte Anfrage lässt sich unter der im "Location"-Header-Feld angegebenen Adresse beziehen. Der Browser soll mit einem GET folgen, auch wenn der ursprüngliche Request ein POST war.
304	Not Modified	Der Inhalt der angeforderten Ressource hat sich seit der letzten Abfrage des Clients nicht verändert und wird deshalb nicht übertragen. Zu den Einzelheiten siehe Browser-Cache-Versionsvergleich.
305	Use Proxy	Die angeforderte Ressource ist nur über einen Proxy erreichbar. Das "Location"-Header-Feld enthält die Adresse des Proxy.
307	Temporary Redirect	Die angeforderte Ressource steht vorübergehend unter der im "Location"-Header-Feld angegebenen Adresse bereit. Die alte Adresse bleibt gültig. Der Browser soll mit derselben Methode folgen wie beim ursprünglichen Request (das heißt, einem POST folgt ein POST).
308	Permanent Redirect	Experimentell eingeführt via RFC; die angeforderte Ressource steht ab sofort unter der im "Location"-Hea- der-Feld angegebenen Adresse bereit, die alte Adresse ist nicht länger gültig.
300	Multiple Choices	Die angeforderte Ressource steht in verschiedenen Arten zur Verfügung. Die Antwort enthält eine Liste der verfügba- ren Arten. Das "Location"-Header-Feld enthält eventuell die Adresse der vom Server bevorzugten Repräsentation.



REST-Webservices



Client-Fehler (Code 4xx)

Code Nachricht Bedeutur		Bedeutung
400	Bad Request	Die Anfragenachricht war fehlerhaft aufgebaut.
401 Unauthorized Die Anfrage durchgeführt führt werden der-Feld der		Die Anfrage kann nicht ohne gültige Authentifizierung durchgeführt werden. Wie die Authentifizierung durchge- führt werden soll, wird im "WWW Authenticate"-Hea- der-Feld der Antwort übermittelt.
403 <i>Forbidden</i> Die Anfrage wurde mange durchgeführt, beispielswei Benutzer nicht berechtigt rierte URL nur mit HTTP a		Die Anfrage wurde mangels Berechtigung des Clients nicht durchgeführt, beispielsweise, weil der authentifizierte Benutzer nicht berechtigt ist oder eine als HTTPS konfigu- rierte URL nur mit HTTP aufgerufen wurde.
<b>404</b> <i>Not Found</i> Die angeforderte Ressource wurde nicht Statuscode kann ebenfalls verwendet we Anfrage ohne näheren Grund abzuweise solche Fehlerseiten verweisen, werden an bezeichnet.		Die angeforderte Ressource wurde nicht gefunden. Dieser Statuscode kann ebenfalls verwendet werden, um eine Anfrage ohne näheren Grund abzuweisen. Links, die auf solche Fehlerseiten verweisen, werden auch als tote Links bezeichnet.
407Proxy Authenti- cationAnalog zum Statuscode 401 ist Authentifizierung des Clients g Proxy erforderlich. Wie die Auth werden soll, wird im "Proxy Aut Antwort übermittelt.		Analog zum Statuscode 401 ist hier zunächst eine Authentifizierung des Clients gegenüber dem verwendeten Proxy erforderlich. Wie die Authentifizierung durchgeführt werden soll, wird im "Proxy Authenticate"-Header-Feld der Antwort übermittelt.
408	Request Time-out	Innerhalb der vom Server erlaubten Zeitspanne wurde keine vollständige Anfrage des Clients empfangen.
409	ConflictDie Anfrage wurde unter falschen Annahmen gestellt. I Falle einer PUT-Anfrage kann dies zum Beispiel auf eine zwischenzeitliche Veränderung der Ressource durch Drit zurückgehen.	



5.5.6.4 Seite 32 Um diese Meldungen auch im HTTP-Bereich entsprechend behandeln zu können, müssen die möglichen HTTP-Ereignisse auch im RPG-Programm definiert werden.

D H_OK	С	const(200)
D H_CREATED	с	const(201)
D H_NOCONTENT	с	const(204)
D H_BADREQUEST	с	const(400)
D H_NOTFOUND	с	const(404)
D H_CONFLICT	с	const(409)
D H_GONE	с	const(410)
D H_SERVERERROR	с	const(500)

Definition von Konstanten

Wenn zum Beispiel versucht wird, einen neuen Kundenstammsatz zu erfassen, den es schon gibt, dann führt das im RPG-Programm zu einem doppelten Schlüssel bzw. im HTTP-Status zu einer Fehlermeldung. Diese weist darauf hin, dass die Neuanlage nicht durchgeführt werden konnte, da der Datensatz bereits existiert.

In diesem Fall würde ein H\_Conflict gesetzt (Fehler 409).

Im Folgenden sind die einzelnen Prozeduren im RPG dargestellt.



#### Neuanlage eines Kundenstammsatzes

Der nachfolgende RPG-Codeausschnitt zeigt die Anweisungen für die Anlage eines neuen Kunden.

```
// create
 P create
              В
                               EXPORT
             ΡI
D create
D Kunde
                               likeds(KundenRDS)
D httpStatus
                         10i <mark>0</mark>
                         100a dim(10)
D httpHeaders
/FREE
 openkundenpf ();
 KundenNr = Kunde.KundenNr;
 VorName = Kunde.VorName;
 NachName = Kunde.NachName;
 KdTyp
        = Kunde.KdTyp;
 write(e) KundenR;
 if NOT %ERROR;
   httpStatus = H_CREATED;
   // URL auf Server und Port anpassen
   httpHeaders(1) = ,Aufruf: , +
           ,http://server:port/web/service/Kunde/' + KundenNr;
 elseif %STATUS = ERR_DUPLICATE_WRITE;
   httpStatus = H_CONFLICT;
 else;
   httpStatus = H_SERVERERROR;
 endif;
 closekundenpf ();
/END-FREE
P create
              Е
```

#### RPG-Codeausschnitt Neuanlage

Bei der Ausführung dieser Prozedur können beispielsweise folgende HTTP-Statuscodes verarbeitet werden:

- 201 Erfolgreiche Neuanlage
- 409 Satz existient bereits
- 500 Server-Fehler





# 5.5.6.4

Seite 34

## Ändern eines Kundenstammsatzes

Der nachfolgende RPG-Codeausschnitt zeigt die Anweisungen für das Ändern eines bestehenden Kundenstammsatzes.

```
// update
 P update
              В
                              EXPORT
             PI
D update
D Kunde
                              likeds(KundenRDS)
D httpStatus
                         10i <mark>0</mark>
 /FREE
 openkundenpf ();
 chain(e) Kunde.KundenNr kundenpf ;
 if (%ERROR);
   httpStatus = H_SERVERERROR;
   return;
 elseif %FOUND;
   KundenNr = Kunde.KundenNr;
   VorName = Kunde.VorName;
   NachName = Kunde.NachName;
   KdTyp = Kunde.KdTyp;
   update(e) KundenR;
   if NOT %ERROR;
    httpStatus = H_NOCONTENT;
   else:
    httpStatus = H_NOTFOUND;
   endif;
 else;
   httpStatus = H_NOTFOUND;
 endif;
 closekundenpf ();
 /END-FREE
P update
              Е
                         :
```

RPG-Codeausschnitt Update

Wird der eingegebene Kundenstammsatz nicht gefunden, dann liefert das Programm den Fehler 404 (h\_NotFound) zurück. Ein erfolgreiches Verändern wird mit dem Status 204 zurückgeliefert.





#### Löschen eines Kundenstammsatzes

Der nachfolgende RPG-Codeausschnitt zeigt die Anweisungen für das Löschen eines bestehenden Kundenstammsatzes.

	//**********	******	*****	**********	
	// Delete			*	
	//*********	******	*****	***********	
Р	Delete	В		EXPORT	
D	Delete	PI			
D	KundenNr		9a	Const	
D	httpStatus		10i <mark>0</mark>		
,	/FREE				
	<pre>openkundenpf ();</pre>				
		la lundonation			
	chain(e) Kundenk	ir kundenpr ;			
	http://www.internationality.com				
	nupstatus = r	I_SERVERERROR;			
	return;				
	elseit %FUUND;				
	delete(e) Kund	ienk;			
	it NOT %ERROR;				
	httpStatus =	<pre>H_NOCONTENT;</pre>			
	elseif NOT %FC	)UND;			
	httpStatus =	<pre>H_NOTFOUND;</pre>			
	else;				
	httpStatus =	H_SERVERERROF	<b>ξ</b> ;		
	endif;				
	else;				
	httpStatus = H	I_NOTFOUND;			
	endif;				
	closekundenpf ()	;			
,	/END-FREE				
Р	Delete	E			

RPG-Codeausschnitt Löschen

Wird der eingegebene Kundenstammsatz nicht gefunden, dann liefert das Programm den Fehler 404 (h\_NotFound) zurück. Ein erfolgreiches Löschen wird mit dem Status 204 zurückgeliefert.





#### Abrufen eines Kundenstammsatzes

5.5.6.4 Seite 36

Der nachfolgende RPG-Codeausschnitt zeigt die Anweisungen für das Abrufen eines Kundenstammsatzes.

	//***********	******	******	********
	// GetByNr			*
	//*********	*******	******	************
Р	GetByNr	В		EXPORT
D	GetByNr	PI		
D	KundenNr		9a	const
D	Kunde			likeds(KundenRDS)
D	httpStatus		10i <mark>0</mark>	
D	httpHeaders		100 <mark>a</mark>	dim(10)
/	FREE			
	clear httpHeader	rs;		
	<mark>clear</mark> Kunde;			
	openkundenpf ()	:		
	chain(e) Kunden	Nr kundenpt ;		
	it (%ERROR);			
	httpStatus = H	1_SERVERERROR	;	
	return;			
	elseit %FOUND;	- Konstantina		
	Kunde Kundenni	r = Kundennr;		
	Kunde.VorName	= vorname;		
	Kunde Katur	e = Nachname;		
	Kunde.Kuryp	= Kulyp;		
	httpStatus - k			
	also:	I_OK,		
	httpStatus - k			
	endif:	I_NOTFOUND,		
	enurr,			
	httpHeaders(1) =	- 'Cache-Conti	rol: no-	-cache, no-store';
	closekundenpf ()	);		
/	END-FREE			
Р	GetByNr	E		

RPG-Codeausschnitt Abrufen Kundenstammsätze



## Abrufen aller Kundenstammsätze

Der nachfolgende RPG-Codeausschnitt zeigt die Anweisungen für das Abrufen aller Kundenstammsätze.

Hier kann eine Liste an Ergebnissen zurückgeliefert werden. Diese wurden mit einem maximalen Vorkommen von 1.000 Sätzen definiert. Würde jetzt nichts weiter angegeben werden, dann würde in jedem Fall das gesamte Array zurückgegeben - auch dann, wenn zum Beispiel nur zwei Kundenstammsätze gefunden würden. Das wäre gerade für solche Anwendungen, die performant ausgeführt werden sollen, unpraktisch! Deshalb wurde in der Schnittstellendefinition der Parameter LENGTH aufgenommen, der die Anzahl der gefundenen Datensätze aufnimmt und diese zurückliefert. Der Webservice erhält damit nur zwei Elemente, und zwar die der beiden gefundenen Kundenstammsätze. Mit dieser kleinen Ergänzung kann das Ergebnis also dahingehend beeinflusst werden, dass nur die Elemente an den Webservice geliefert werden, die auch tatsächlich gefüllt sind.



Seite 37



**REST-Webservices** 

5.5.6.4 Seite 38

```
// getAll
 P getAll
               В
                                EXPORT
D getAll
              PI
D Kunde_...
D LENGTH
                           10i 0
D Kunde
                                 likeds(KundenRDS) dim(1000)
D
                                 options(*varsize)
D httpStatus
                           10i <mark>0</mark>
D httpHeaders
                          100a dim(10)
 /FREE
 clear httpHeaders;
 clear Kunde;
 Kunde_LENGTH = 0;
 openkundenpf ();
 setll *loval kundenpf ;
 read(e) KundenR;
 if (%ERROR);
   httpStatus = H_SERVERERROR;
   return;
 endif;
 dow (NOT %eof);
   Kunde_LENGTH = Kunde_LENGTH+1;
   Kunde(Kunde_LENGTH).KundenNr = KundenNr;
   Kunde(Kunde_LENGTH).VorName = VorName;
   Kunde(Kunde_LENGTH).NachName = NachName;
   Kunde(Kunde_LENGTH).KdTyp = KdTyp;
   read(e) KundenR;
   if (%ERROR);
    httpStatus = H_SERVERERROR;
     return;
   endif;
 enddo;
 httpStatus = H_OK;
 httpHeaders(1) = 'Cache-Control: no-cache, no-store';
 closekundenpf ();
 /END-FREE
P getAll
               Е
```




### 5.5.6.4.1 Installation des Webservices

Basierend auf den RPG-Prozeduren und dem Serviceprogramm kann nun der REST-Webservice bereitgestellt werden.

Dazu ist es erforderlich sich an der HTTP-Admin-Konsole anzumelden und die Option "Neuen Webservice implementieren" zu wählen.

In der nächsten Seite wird als Webservice-Typ REST ausgewählt.

**REST-Webservice** installieren

Ergänzung 43/2017





IBM Web Administration for i Setup Manage Advanced | Related Links (?) WebSphere. IBM All Servers | HTTP Servers Application Servers Installations Server: WSERVICE - V2.6 (web services) ✓ Common Tasks and Wizards
 Create Web Services Server
 Create HTTP Server
 Create Application Server
 Create WebSphere Portal WSERVICE > Manage Deployed Services > Deploy New Service Deploy New Service Specify Web service type - Step 1 of 9 Web Services Wizards Welcome to the Deploy New Service wizard. This wizard helps you externalize an IBM i program object as a Web service. Deploy New Service Configure SSL Disable SSL Specify Web service type: 📀 OSOAP Server Properties
 Properties
 Server Tracing
 View HTTP Servers A SOAP-based Web service is a self-contained software component with a well-defined interface that describes a set of operations sible over the Internet and exchange XML messages that are based on the SOAP protocol. REST A REST-based Web service exposes resources, where client requests are handled by resource methods and the format of messages that are exchanged is defined by the resource itself. Problem Determination View Logs
 View Log Monitor
 Point Create Summary Back Next Cancel 



## Web-Programmierung

**REST-Webservices** 

5.5.6.4 Seite 40 Nun muss das RPG- oder Serviceprogramm angegeben werden, das mit dem Webservice genutzt werden soll.



#### IBM-i-Objekte

Basierend auf den PCML-Informationen können später Parameterdefinitionen hinterlegt werden.

Doch vorher muss dem Webservice ein Name gegeben werden, unter dem dieser im Administrationsbereich ersichtlich wird. Hier lautet er KUNDENWS.

IBM Web Administration for Setup Manage Advanced	i I   Related Links		(?) WebSphere.	IBM
All Servers   HTTP Servers	Application Servers	Installations		
🛚 🕒 🔽 🚱 Ser	ver: WSERVICE - V2	.6 (web services) 🗸		
<ul> <li>Common Tasks and Wizards</li> <li>Create Web Services Server</li> <li>Create HTTP Server</li> <li>Create Application Server</li> <li>Create WebSphere Portal</li> </ul>	WSERVICE > M Deploy Nev Specify Name	anage Deployed Services > Deploy New Service <b>v Service</b> for Service - Step 3 of 9		
Web Services Wizards     Deploy New Service     Configure SSL     Disable SSL	The Web servic path is relative further restrict v	e to be externalized is a resource. The URI path template identifies matching patterns fo o the context root and can be a simple string or one or more template parameters that c shat is allowed.	r incoming HTTP requests an contain regular express	s. The sions to
Server Properties     Properties     Server Tracing     View HTTP Servers	Resource i Service des URI path te	ame: KUNDENWS cription: WebService Kundenstamm X mplate: / e.g. /temperature, /temperature/(temp?	d+}	
Services     Manage Deployed Services     Problem Determination     Niew Logs				
4 Web Log Monitor "≫ View Create Summary				
	Back Next	Cancel		





Weh-Proar	ammierund
TCD-I I UQI	anniciany

apite

Seite 41

Basierend auf dem RPG-Serviceprogramm werden nun die verschiedenen Prozeduren vorgeschlagen.

An dieser Stelle wird jeder einzelne Eintrag erweitert, um die Parameterart (Eingabe/Ausgabe) festzulegen.

IBM Web Administration for i	lated Links					WebSphere,	IBM
All Servers   HTTP Servers Appli	cation Serve	ers Installations					
🛚 Running 🗋 🔲 🛃 Server: 🚺	WSERVICE	- V2.6 (web services) 🗸					
<ul> <li>✓ Common Tasks and Wizards</li> <li><sup>1</sup> Greate Web Services Server</li> <li><sup>1</sup> Greate HTTP Server</li> <li><sup>1</sup> Greate Application Server</li> <li><sup>1</sup> Greate WebSphere Portal</li> </ul>	procedure clients and performan	row to change the default settings for the pr d what is returned by the Web service. For ar ce.	rocedure parai rray type paran	neters. The Us neters, modifyi	sage parameter attribute aff ng the Count field may impr	ects what data is se ove Web service	nt by
- Web Services Wizards	Export pro	cedures: 🔞					
Deploy New Service	Select	Procedure name/Parameter name	Usage	Data type	Count		
Disable SSL	✓	▼ DELETE					
- Server Properties		KUNDENNR	input 🗸	char			
h Properties		HTTPSTATUS	output 🗸	int			
B Server Tracing	~	▼ UPDATE					
- Sanicas		KUNDE	input 🗸	struct			
Manage Deployed Services		HTTPSTATUS	output 🗸	int			
<ul> <li>Problem Determination</li> </ul>	~	▼ CREATE					
View Logs     Web Log Monitor		KUNDE	input 🗸	struct			
P View Create Summary		HTTPSTATUS	output 🗸	int			
		HTTPHEADERS	output 🗸	char	10 🗸		
	~	GETBYNR					·
	Back	Cancel					

Exportprozeduren

Um die Anzahl der flexiblen Array-Inhalte beeinflussen zu können, wird für die Prozedur GETALL der Zählerwert KUNDE\_LENGTH angegeben.

IBM Web Administration for i						(?) WebSphere.	IBM
Setup Manage Advanced Rela		- Installations					
All Servers   HTTP Servers   Applic	ation Servers	Installations					
🗣 Running 🚺 🛄 🛃 Server: 🕅	SERVICE -	V2.6 (web services) V					
Common Tasks and Wizards     Create Web Services Server		- CREATE	_ output ·				^
1 Create HTTP Server 1 Create Application Server 1 Create WebSphere Portal	1	KUNDE	input 🗸	struct			
	1	HTTPSTATUS	output 🗸	int			
<ul> <li>Web Services Wizards</li> </ul>	1	HTTPHEADERS	output 🗸	char	10 🗸		
Deploy New Service	<ul> <li>✓</li> </ul>	▼ GETBYNR					
Disable SSL	1	KUNDENNR	input 🗸	char			
	1	KUNDE	output 🗸	struct			
Properties     Server Tracing	1	HTTPSTATUS	output 🗸	int			
View HTTP Servers		HTTPHEADERS	outout 🗸	char	10 🗸		
✓ Services	<b>v</b>	▼ GETALL					
Manage Deployed Services	1	KUNDE_LENGTH	output 🗸	int			
Problem Determination	1	KUNDE	input 🗸	struct	KUNDE_LENGTH V	]	
4 Web Log Monitor		HTTPSTATUS	output 🗸	int			
P View Create Summary	· ·	HTTPHEADERS	output 🗸	char	10 🗸	J	
	Select All	Deselect All Expand All Collapse	All				$\sim$
	Back Nex	t Cancel					

Ergänzung 43/2017

Array-Angaben

Nach der Festlegung der Parameterbehandlung führt der Assistent durch die Konfiguration der HTTP-Angaben für die einzelnen Prozeduren.



## 5.5.6.4 Seite 42

#### 5.5.6.4.2 Webservice Kunden löschen

Nachfolgend finden Sie die Einstellungen für das Löschen eines Kundenstammsatzes mit dem REST-Webservice.

Als Schlüsselwert für den Aufruf muss die Kundennummer des zu löschenden Datensatzes angegeben werden. Dazu wird der Parameter KD definiert.

Achten Sie darauf, dass als Prozedurname DELETE voreingestellt ist.

- Wählen Sie im Feld "HTTP request method" DELETE aus.
- Geben Sie im Feld "URI path template for method" den Wert {kd} an.
- Wählen Sie im Feld "HTTP response code output parameter" HTTPSTATUS.
- Wählen Sie im Feld "Returned output media types" \*JSON.
- Selektieren Sie im Bereich "Input parameter mappings" als Input Source \*PATH\_PARAM und achten Sie darauf, dass als Identifier KD eingestellt ist.

IBM Web Administration for i		WebSphere	TBM
Setup Manage Advanced	Related Links	webophere,	ieiii
All Servers   HTTP Servers A	pplication Servers Installations		
🛛 🕞 🔽 🚱 Serve	er: WSERVICE - V2.6 (web services) 🗸		
<ul> <li>Common Tasks and Wizards</li> <li>Create Web Services Server</li> <li>Create HTTP Server</li> </ul>	Specify resource method information.		^
Create Application Server	Procedure name: DELETE		
Create webspilere Poltar	URI path template for resource: /		
<ul> <li>Web Services Wizards</li> </ul>	HTTP request method: DELETE V		
Deploy New Service	URI path template for method: {kd} or V		
🖞 Disable SSL	HTTP response code output parameter. HTTPSTATUS 🗸		
<ul> <li>Server Properties</li> </ul>	HTTP header array output parameter: *NONE 🗸		
B Properties B Server Tracing	Allowed input media types: *ALL or 🗸		
View HTTP Servers	Returned output media types: *JSON or V		
▼ Services	Whether to wrap input parameters:		
B Manage Deployed Services	O Wrap input parameters		
<ul> <li>Problem Determination</li> </ul>	Do not wrap input parameters		
View Logs     View Log Monitor	Input parameter mappings:		
P View Create Summary	Parameter name Data type Input source Identifier	Default Value	
	KUNDENNR char *PATH_PARAM V kd V *NONE	or 🗸	
			$\sim$
	Back Next Cancel		
			^

Ressourcenmethoden Informationen

Mit einem Klick auf die Schaltfläche "Next" gelangen Sie zum Definitionsbereich für die nächste Prozedur.



## 5.5.6.4.3 Webservice Kunden-Update

Für das Kunden-Update verwenden Sie die HTTP-Request-Methode PUT.

Da als Eingabe/Ausgabe JSON verwendet wird, müssen die Auswahlen auch im Definitionsbereich beschränkt werden, indem in den Feldern "Allowed input media types" und "Returned output media types" \*JSON selektiert wird.

IBM Web Administration for i								(2) WebSphere	TRM
Setup Manage Advanced	Related Lir	iks							
All Servers   HTTP Servers	Application S	ervers	Installations						
👻 Running 🕞 🔲 💋 Serv	er: WSERV	'ICE - V2	.6 (web services	) 🗸					
Common Tasks and Wizards     Create Web Services Server     Create HTTP Server     Create Application Server	Specif	y resourc	e method informa	ition. 😮					^
Create WebSphere Portal	Pr	ocedure r	name:	L. L.	UPDATE				
Web Ore free Witnesde	UF	RI path te	mplate for resour	ce: /	1				
Vied Services Wizards     Deploy New Service	H	FTP reque	est method:		PUT 🗸				
Disable SSL	UF	RI path te	mplate for metho	t [	*NONE	0	r 🗸		
	H	ITP respo	onse code output	parameter: [	HTTPSTATUS 🗸	l			
<ul> <li>Server Properties</li> <li>Properties</li> </ul>	H	FTP head	ier array output pa	rameter: [	*NONE V				
Server Tracing     New HTTP Servers	All	owed inp	ut media types:		*JSON	0	r 🗸		
	R	eturned o	utput media types	c i	*JSON	0	r 🗸	1	
▼ Services	w	hether to	wrap input param	eters:				•	
manage Deployed Services		() Wrap	input parameters						
<ul> <li>Problem Determination</li> </ul>		Do no	nt wran input nara	meters					
View Logs     Web Log Monitor		lanut							
View Create Summary		Dar	parameter mappi	ngs. Data tupo		Identifie	Ar Dofault Value	1	
		Par KUI		etruct	*NONE	Tuenum			
		KU		anuct	NONE			]	$\sim$
	Back	Novt	Cancol						_
	DdCK	Next	calicer						
									^

### JSON-Angaben

Mit einem Klick auf die Schaltfläche "Next" gelangen Sie zum Definitionsbereich für die nächste Prozedur.



5.5.6.4

Seite 43

Ergänzung 43/2017

## 5.5.6.4 Seite 44

#### 5.5.6.4.4 Webservice Kunde erstellen

Für die Neuanlage eines Kundenstammsatzes wird die HTTP-Methode POST verwendet.

Auch hier soll sich auf JSON-Ein- und Ausgabewerte beschränkt und zudem angegeben werden, dass die HTTP-Methode über den HTTPSTATUS den Verarbeitungsstatus liefert und über HTTPHEADERS der Erfolg der Neuanlage quittiert wird.

IBM Web Administration for i Setup Manage Advanced	kelated Links	WebSphere.	IBM
All Servers   HTTP Servers Ap	Dication Servers Installations		
🛚 Running 🕞 🔲 💋 Server	WSERVICE - V2.6 (web services) V		
<ul> <li>Common Tasks and Wizards</li> <li>Create Web Services Server</li> <li>Create HTTP Server</li> <li>Create Application Server</li> <li>Create WebSphere Portal</li> </ul>	request method to a resource method. Specify resource method information.		^
	Procedure name: CREATE		
<ul> <li>Web Services Wizards</li> <li>Deploy New Service</li> </ul>	URI path template for resource: /		
Configure SSL	HTTP request method: POST V		
UISADIE SOL	URI path template for method:		
<ul> <li>Server Properties</li> </ul>	HTTP response code output parameter: HTTPSTATUS 🗸		
B Properties B Server Tracing	HTTP header array output parameter: HTTPHEADERS V		
B View HTTP Servers	Allowed input media types: *JSON or V		
▼ Services	Returned output media types: *JSON or V		
B Manage Deployed Services	Whether to wrap input parameters:		
<ul> <li>Problem Determination</li> </ul>	O Wrap input parameters		
4 Web Log Monitor	O not wrap input parameters		
P View Create Summary	Input parameter mappings:		
	Parameter name Data type Input source Identifier Default Value		<u> </u>
	KUNDE struct *NONE		· ·
	Back Next Cancel		

### HTTPSTATUS

Mit einem Klick auf die Schaltfläche "Next" gelangen Sie zum Definitionsbereich für die nächste Prozedur.



## 5.5.6.4.5 Webservice Kundenabfrage einzeln

Informationen zu einem einzelnen Kunden werden mit GETBYNR abgerufen. Dazu nutzen Sie die HTTP-Methode GET und verwenden wieder als Schlüsselwert die Kundennummer. Den Verarbeitungsstatus und weitere Informationen rufen Sie auch hier mit HTTPSTATUS und HTTPHEADERS ab.

Achten Sie darauf, dass Sie in den Eingabe-Sourcen \*PATH\_PARAM angeben und der Identifier mit KD gesetzt ist.

IBM Web Administration for i Setup Manage Advanced   Rel	lated Links	IBM
All Servers   HTTP Servers Appli	cation Servers Installations	
🛚 Running 🔃 🔲 💋 Server: 🛛	VSERVICE - V2.6 (web services) V	
✓ Common Tasks and Wizards <sup>1</sup> <sup></sup>	Specify resource method information. 2	^
Create Application Server	Procedure name: GETBYNR	
Create WebSphere Portai	URI path template for resource: /	
✓ Web Services Wizards	HTTP request method: GET 🗸	
Deploy New Service	URI path template for method: {kd} or V	
🖞 Disable SSL	HTTP response code output parameter: HTTPSTATUS 🗸	
Server Properties	HTTP header array output parameter: HTTPHEADERS V	
B Server Tracing	Allowed input media types: *JSON or V	
b View HTTP Servers	Returned output media types: *JSON or V	
▼ Services	Whether to wrap input parameters:	
B Manage Deployed Services	O Wrap input parameters	
<ul> <li>Problem Determination</li> </ul>	Do not wrap input parameters	
View Logs     Web Log Monitor	Input parameter mappings:	
P View Create Summary	Parameter name Data type Input source Identifier Default Value	
	KUNDENNR char *PATH_PARAM V kd V *NONE or V	
		· ·
	Back Next Cancel	

## PATH\_PARAM

Mit einem Klick auf die Schaltfläche "Next" gelangen Sie zum Definitionsbereich für die nächste Prozedur.



5.5.6.4

Seite 45

Ergänzung 43/2017



5.5.6.4 Seite 46

## 5.5.6.4.6 Webservice Kundenabfrage gesamt

Den gesamten Kundenstamm rufen Sie mit der Methode GETALL ab. Auch hier wird die HTTP-Methode GET verwendet – diesmal aber ohne jeglichen einschränkenden Parameter.

IBM Web Administration for i Setup Manage Advanced   Rel	ated Links		WebSphere.	BM
All Servers   HTTP Servers Appli	cation Servers Installations			
🛚 Running 📄 🔲 💋 Server: 🛛	VSERVICE - V2.6 (web services) 🗸			
Common Tasks and Wizards        Create Web Services Server       Create HTTP Server       Create Application Server       Create WebShares Portal	WSERVICE > Manage Deployed Services > D Deploy New Service Specify Resource Method Information - Sh	eploy New Service		
▼ Web Services Wizards <sup>1</sup> ∂ Deploy New Service <sup>1</sup> ∂ Configure SSL <sup>1</sup> ∂ Disable SSL	Procedures are mapped to resource method request method to a resource method.	ds. Each resource method n	needs to be defined to handle client requests by mapping an H1	TTP
<ul> <li>✓ Server Properties</li> <li>B Properties</li> <li>B Server Tracing</li> <li>D View HTTP Servers</li> </ul>	Procedure name: URI path template for resource:	GETALL		
▼ Services	HTTP request method: URI path template for method:	GET V *NONE	or V	
<ul> <li>Problem Determination</li> <li>View Logs</li> <li>Web Log Monitor</li> <li>View Create Summary</li> </ul>	HTTP response code output parameter: HTTP header array output parameter: Allowed input media types:	HTTPSTATUS V		
	Returned output media types:	*JSON		
	Back Next Cancel		1	-

#### GETALL

Damit wurden alle Methoden definiert.

Jetzt muss nur noch angegeben werden, unter welchem Benutzerprofil das RPG-Programm ausgeführt werden soll. Hier müssen Sie ein IBM-i-Benutzerprofil selektieren, das über die erforderlichen Ausführungsberechtigungen auf dem System i verfügt.

IBM Web Administration for i	elstard Links	IBM
All Servers   HTTP Servers Appl	lication Servers Installations	
🛚 Running 🗋 🔲 💋 Server:	WSERVICE - V2.6 (web services) V	
<ul> <li>▼ Common Tasks and Wizards</li> <li>♥ Create Web Services Server</li> <li>♥ Create Application Server</li> <li>♥ Create Application Server</li> <li>♥ Create WebSphere Portal</li> </ul>	WSERVICE > Manage Deployed Services > Deploy New Service Deploy New Service Specify User ID for this Service - Step 6 of 9	
✓ Web Services Wizards <sup>1</sup> Deploy New Service <sup>1</sup> Configure SSL <sup>1</sup> Disable SSL	The service requires an IBM i user ID to run the program object that contains the Web service business logic. Specify User ID for this Service: 🚱	
<ul> <li>✓ Server Properties</li> <li>△ Properties</li> <li>△ Server Tracing</li> <li>△ View HTTP Servers</li> </ul>	<ul> <li>Use server's user ID</li> <li>Specify an existing user ID</li> <li>The user ID must have the necessary authority to this program object and any other additional program objects.</li> </ul>	
▼ Services	User ID: <u>∠EIG</u> ✓ Update the server's user ID to have *USE authority to this user ID.	
<ul> <li>              Problem Determination             D View Logs             4             Web Log Monitor             <sub>g</sub><sup>b</sup> View Create Summary      </li> </ul>		
	Back Next Cancel	



Benutzerangaben

	REST-Webser
In der nächsten	Anzeige können Sie bei Bedarf die Bibliotheksliste anpassen.
damit dag gufr	unufondo DDC Drogramm auch konnelet auggeführt worden
danni das aurz	uruiende RPG-Programm auch korrekt ausgeführt werden
kann.	
Für dieses Beispi	el wird keine weitere Bibliothek benötigt. Daher erfolgt auf der
Anzeige ein Klic	k auf "Next".
IBM Web Administration for i	(?) WebSphere,
Setup Manage Advanced F	elated Links
Running Running Servers	
Common Tasks and Wizards	WSERVICE > Manana Daniavad Sanirae > Daniav New Sanira
Create Web Services Server	TOLITICE - manage septered entrops
Create Application Server	Deploy New Service
Create WebSphere Portal	Specify Library List - Step 7 of 9
<ul> <li>Web Services Wizards</li> <li>Deploy New Service</li> </ul>	The functionality of the IBM i program you want to externalize as a Web service may depend upon other IBM i programs in the system.
Configure SSL	Specify all libraries in which programs exist that the Web service programs depend on. If no library is specified, a default library list is used.
Disable SSL	Specify library list position for this Web service:
<ul> <li>Server Properties</li> <li>Properties</li> </ul>	○ Insert libraries in front of user library portion of the library list
Server Tracing     View HTTP Servers	Insert libraries at the end of user library portion of the library list
	Library list entries: 🕑
<ul> <li>Services</li> <li>Manage Deployed Services</li> </ul>	
- Broblem Determination	ádd Remove áll
► View Logs	
4 Web Log Monitor P View Create Summary	
	Back Next Cancel

## Bibliotheksliste

Als Transport-Metadata wird REMOTE\_ADDR ausgewählt.

IBM Web Administration for	WebSnhere	đv
Setup Manage Advanced	I   Related Links	91N
All Servers   HTTP Servers	Application Servers Installations	
🛚 Running 🚺 🔲 🗭 Ser	ver: WSERVICE - V2.6 (web services) 🗸	
✓ Common Tasks and Wizards <sup>1</sup> Create Web Services Server <sup>1</sup> Create HTTP Server	Information to be passed to web service implementation code ${f Q}$	^
Create Application Server	Specify Transport Metadata:	
-	Transport Metadata	
<ul> <li>Web Services Wizards</li> <li>Deploy New Service</li> </ul>	QUERY_STRING	- 14
2 Configure SSL	REMOTE_ADDR	
1 Disable SSL	REMOTE_USER	
✓ Server Properties	REQUEST_METHOD	
Properties     Server Tracing		
D View HTTP Servers	REQUEST_URL	
- Services	SERVER_NAME	
Manage Deployed Services	SERVER_PORT	
	Specify HTTP Headers:	
4 Web Log Monitor View Create Summary	HTTP Headers	
	There are no entries for this table.	
	Add Remove All	$\sim$
	Back Next Cancel	

## Transport

Kapitel

5.5.6.4 Seite 47

Web-Programmierung



## Web-Programmierung

**REST-Webservices** 

## 5.5.6.4 Seite 48

Damit ist die Definition des Webservices abgeschlossen. In der angezeigten Zusammenfassung können die getätigten Angaben geprüft werden. Der Webservice für die Kundenstammverwaltung wird dann mit einem Klick auf "Fertig stellen" auf dem Webserver angelegt.

IBM Web Administration for i	stadilinke	? WebSphere,	IBM
All Servers   HTTP Servers Applie	cation Servers Installations		
🗣 Running D 🔲 🧭 Server: V	VSERVICE - V2.6 (web services) V		
Common Tasks and Wizards     Create Web Services Server     Create HTP Server     Create Application Server     Create WebSphere Portal     Web Services Wizards	WSERVICE > Manage Deployed Services > Deploy New Service Deploy New Service Summary - Step 9 of 9		
안 Deploy New Service 안 Configure SSL 안 Disable SSL ▼ Server Properties	Service Methods Request Information		
D Properties D Server Tracing D View HTTP Servers Services D Manage Deployed Services	Resource name:         KUNDENWS           Resource description:         Kunden REST WebService           Service install path:         /www/wservice/webservices/kUNDENWS           URI path template:         /           User ID for service:         ZEIG		
Problem Determination     D View Logs     Web Log Monitor     J <sup>b</sup> View Create Summary	Program: /QSYS.LIB/ZEIG.LIB/KUNDENWS.SRVPGM Library list for service: ZEIG		
	Back Finish Cancel		

#### Zusammen

Nach kurzer Zeit sollte der neue Webservice KUNDENWS in der Auflistung der installierten Webservices als "aktiv" erscheinen.

IBM Web Administration for i Setup Manage Advanced   Related Links							
All Servers   HTTP Servers Application Servers Installations							
🛯 Running 🔃 🛄 💋 Server: 🚺	Running D D Server: WSERVICE - V2.6 (web services) V						
Common Tasks and Wizards     WSERVICE > Manage Deployed Services     Create HTTP Server     Create Application Server     Manage Deployed Services							
✓ Web Services Wizards ♥ Deploy New Service	Data current as of May 8, 20 Deployed services: ②	017 1:09:17 PI	М.				
한 Configure SSL 한 Disable SSL	Service name     KUNDENWS	Status Sunning	Type REST	Startup type Automatic	Service definition		

#### Webservice Laufstatus

Um die Eigenschaften des Webservices betrachten zu können, wird er zunächst markiert und anschließend auf die Schaltfläche "Eigenschaften" geklickt.

Hier besteht Zugang zu den Einstellungen des Webservices – unter anderem ist dort auch die Aufruf-URL für den Webservice zu finden. Diese wird später für den Test bzw. den Aufruf des Kunden-Webservices benötigt.

Im Beispiel ist dies die URL http://i5ZeBIS:10010/web/services/KUNDENWS.



<ul> <li>Im Vergleich zum SOAP-Webservice kann der REST-Webservice nicht Hilfe des Test-Clients getestet werden. Nichtsdestotrotz kann er mittels ei Browsers nun durchgeführt werden.</li> <li>Beispielhaft sei der Abruf des gesamten Kundenstamms aufgezeigt.</li> <li>Dazu wird ein beliebiger Browser geöffnet und als URL der Link auf den W service angegeben (http://i5ZeBIS:10010/web/services/KUNDENWS).</li> </ul>	mit nes eb-
Beispielhaft sei der Abruf des gesamten Kundenstamms aufgezeigt. Dazu wird ein beliebiger Browser geöffnet und als URL der Link auf den W service angegeben (http://i5ZeBIS:10010/web/services/KUNDENWS).	eb-
Dazu wird ein beliebiger Browser geöffnet und als URL der Link auf den W service angegeben (http://i5ZeBIS:10010/web/services/KUNDENWS).	eb-
Image: Second Secon	
rite tolit view Pavontes Tools Piep	÷
	_
Do you want to open or save KUNDENWS.json (254 bytes) Open Save 🔻 Cancel 🗙	

Webservice Test

Nachdem der Link eingegeben und abgesendet worden ist, erscheint ein Hinweis, dass eine Datei KUNDENWS.JSON geöffnet werden kann. Dabei handelt es sich bereits um die Kundenauflistung im JSON-Format.

Diese wird abgespeichert und anschließend mit einem geeigneten Editor (zum Beispiel RDi) geöffnet.

Das Ergebnis ist die Auflistung des Inhalts der Kundenstammdatei.

```
🗟 KUNDENWS.json 🔀
                                                                                                                                                   {"KUNDE_LENGTH":3, "KUNDE":[{"KUNDENNR": "1000", "VORNAME": "Willi", "NACHNAME": "Winzig", "KDTYP": "Endkunde"}, {"KUNDENNR": "1001", "VORNAME": "Maj
```

## Ergebnisanzeige JSON

Es wurde ein lauffähiger und universell einsetzbarer REST-Webservice generiert.

apitel

Seite 49



Web-Programmierung REST-Webservices

5.5.6.4 Seite 50





Seite 1

## 5.5.7 SOAP Webservices

Die Welt der Webservices verwendet unterschiedliche Standards, so zum Beispiel neben dem bereits gezeigten REST Webservice der SOAP Webservice.

SOAP steht für "Simple Object Access Protocol", wobei die Bezeichnung "Simple" hier als irreführend angesehen werden kann, denn diese Form der Webservices ist in deren interner Behandlung und im Vergleich zu REST alles andere als simpel.

Dennoch werden die SOAP Webservices häufig eingesetzt. Doch letztlich ist die Entscheidung, ob ein REST oder ein SOAP Webservice erstellt werden soll, davon abhängig, welchen der beiden Standards der Partner unterstützt. So ist es durchaus möglich, dass auf IBM-i-Seite ein Webservice im SOAP-Bereich erstellt werden muss, der dann in eine externe Anwendung einzubinden ist.

Die Voraussetzungen aufseiten des RPG-Programms sind dabei ähnlich wie im REST-Umfeld.

In Ihrem Fall nehmen Sie die Differenzierung, ob es sich um einen REST oder einen SOAP Webservice handeln soll, bei der Konfiguration des Webservices selbst vor. Nachfolgende Abbildung zeigt die mögliche Selektion, die allerdings nur bei Servern neuerer Art zur Auswahl steht.

## Hinweis:

Ältere Varianten des Webservers auf dem System i unterstützen nur die SOAP Webservices. Sollten Sie also in der Anzeige keine Auswahl zwischen SOAP und REST finden, dann deshalb, weil Sie noch eine Altversion des HTTP-Servers auf System i verwenden. Das soll aber für das Arbeiten mit den SOAP Webservices zunächst nicht stören. Allerdings können Sie sich in dem Fall auch nicht der Neuerungen bedienen, die IBM mit den neuen Webservern und den in den RPG-Komponenten enthaltenen Erweiterungen bereitgestellt hat. Kapitel

## Web-Programmierung

SOAP Webservices

## 5.5.7 Seite 2

A (1) (10.1.2.125:2001/)	HTTP/ ク ~ 目 0 (④101.2.125 - IBM Navigator for i   億 HTTP-Server-Administratio × ① ☆ ③
File Edit View Favorites Too	ls Help
IBM Web Administration for i	WebSnbere
Konfiguration Verwaltung E	invertert   Zugehörige Links
Alle Server   HTTP-Server Appl	lication Server Installationen
🛛 🕒 Gestartet 📄 🔽 🧭 Server:	WSERVICE - V2.6 (Web-Services) V
<ul> <li>Allgemeine Tasks und Assistenten</li> <li>Web Services Server erstellen</li> <li>UTTR Server erstellen</li> </ul>	WSERVICE > Implementierte Services verwalten > Neuen Service implementieren
· Anwendungsserver erstellen · WebSphere Portal erstellen	Neuen Service implementieren Soecify Web service type - Schritt 1 yon 9
✓ Web-Service-Assistenten <sup>1</sup> Neuen Service implementieren <sup>1</sup> SSL konfigurieren <sup>1</sup> Disable SSL	Willkommen beim Assistenten für die Implementierung eines neuen Service. Dieser Assistent unterstützt Sie beim Auslagern eines IBM i-Programmobjekts als Web-Service.
<ul> <li>✓ Server-Eigenschaften</li> <li>△ Eigenschaften</li> <li>△ Server-Tracing</li> <li>△ HTTP-Server anzeigen</li> </ul>	Typ des Web-Service angeben:
✓ Services	von Arbeitsgängen beschreibt, die über das Internet zugänglich sind und XML-Nachrichten austauschen, die auf dem SOAP-Protokoll basieren.
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> <li>Web Log Monitor</li> <li><sup>ab</sup> Übersicht anzeigen</li> </ul>	Ein REST-basierter Web-Service stellt Ressourcen bereit, wobei Clientanforderungen von Ressourcenmethoden verarbeitet werden und das Format der ausgetauschten Nachrichten von der Ressource selbst definiert wird.
	Zurück Waiter Abbrechen

Auswahl des Webservice-Typs

Die Bereitstellung eines ILE- oder eines Serviceprogramms (unterstützt werden RPG und Cobol) als Webservice kann direkt bei der Neuanlage eines Servers oder auch später erfolgen. Für dieses erste Beispiel übernehmen Sie die Standardeinstellung, wie sie in der folgenden Abbildung zu sehen ist. In Abhängigkeit der hier getätigten Auswahl variieren die Folgeanzeigen. Nach der Übernahme der Standardwerte setzen Sie die Konfiguration mit einem Klick auf "Weiter" fort.



## Web-Programmierung

SOAP Webservices





IBM-i-Programm als Webservice auslagern



5.5.7 Seite 4

#### 5.5.7.1 Details zum IBM-i-Programm

Für die Verwendung der IBM-i-Programme eignen sich ILE-RPG- oder Cobol-Programme. Als Webservice einzubindende Programme müssen entsprechend vorbereitet und angepasst werden. Das ist jedoch kein Problem, denn die notwendigen Anpassungen beschränken sich auf einige wenige Codezeilen, wie Sie im Folgenden sehen werden.

Die Anbindung eines ILE-Programms als Webservice erfolgt auf Basis von PCML-Dokumenten. PCML steht für "Program Call Markup Language" und ist eine in XML erzeugte Beschreibung für die Ein- und Ausgabeparameter des Services, der diese ans aufrufende Programm übergeben muss. Die für die Verwendung von PCML notwendigen Informationen werden vom Konfigurationsassistenten abgerufen und entsprechend gespeichert. Bei der Neuanlage eines Services müssen die Parameter einmalig angegeben werden, doch können sie bei Bedarf auch später angepasst werden.

Für die Erstellung der PCML-Informationen dürfen Sie sich unterschiedlicher Werkzeuge bedienen:

- 1. Sie können einen Texteditor einsetzen.
- 2. Sie können RDp for IBM i verwenden.
- 3. Sie können das ILE-Modul kompilieren. Damit werden die PCML-Informationen direkt ins Modulobjekt eingebunden und können von dort verwendet werden.

Die PCML-Informationen werden bei der Kompilierung über den Parameter Program Interface Information (PGMINFO) gesteuert. Dieser Parameter steht für die Befehle CRTRPGMOD und CRTCBLMOD zur Verfügung.

Im folgenden Beispiel wurde mit RDp kompiliert. Natürlich steht der PCML-Parameter auch bei der Wandlung mit PDM-Funktionen zur Verfügung. Da jedoch Webservices entwickelt werden sollen, bietet sich der konsequente Einsatz eines Werkzeugs an, mit dem alle Komponenten der modernen Anwendungsentwicklung genutzt werden können.



O PDC Modul orstollon (CPTPDCMOD)		
vioaui ersetzen:	> 'YES	
Berechtigung:	*LIBCRTAUT	Name
Vumerische Daten abschneiden:	*YES 🗸	
Numerische Daten korrigieren:	Hipzufi	ligen
7		John
Liei-Release:		
Nullwerte zulassen: Redingungsnamen definieren:		
eungungsnamen dennieren.		Minzufügen
eistungsdatenerfassung aktiv.:	*PEP 🔽	
Profildefinitionsdaten:	*NOCOL	
Binderverzeichnis:	*NONE	Name
Bibliothek:	*LIBL V Hinzufüge	Name
76 O-10		Zaiak an unauk
IC-Optionen:		Zeichenwert
larraichaic ainschliaßan.		
erzeichnis einschlieben;		Minzufügen
rogSchnittstelleninfo:		
Generieren:	*PCML V	
Position:	*MODULE 💌	
		<u>^</u>
atenstromdatei f. ProgSchn.:		
		✓
orprozessoroptionen:	Hinzu	ıfügen
usqabequellepdatei:		Name
Bibliotbek:		Name
<		>
Erweitert Alle Parameter Sch	nlüsselwörter	
CRTRPGMOD MODULE(ZEIG/QIWSSAMPLE) SRC DBGVIEW(*SOURCE) REPLACE(*YES) PGMINFO	FILE(ZEIG/QRPGLESRC) SRCMBR(QIWSSAMPLE) C (*PCML *MODULE)	PTION(*EVENTF)
	OK Chandauduuuta usia dashasatal	
	Diandardwerte wiederherstei	Aubrechen

PCML-Informationen

RPG hat dieses Beispiel als Modul ausgeliefert, weshalb für das Kompilieren der Befehl CRTRPGMOD verwendet wird. Natürlich kann der Programmcode auch so angepasst werden, dass direkt ein Programmobjekt erstellt wird. Doch wurde an dieser Stelle bewusst auf das Anpassen des Codes verzichtet. apitel

5.5.7 Seite 5

Web-Programmierung

SOAP Webservices



Web-Programmierung

**SOAP Webservices** 

5.5.7 Seite 6 Damit der Code als Programm ausgeführt werden kann, erstellen Sie als Zwischenschritt ein ausführbares Programm. Dazu verwenden Sie den Befehl CRTPGM.

©] Sitzung A - [24 x 80]	
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hilfe	
Programm erstellen (CRTPGM)	
Auswahl eingeben und Eingabetaste drücken.	
Programm        giwssample       Name         Bibliothek        zeiq       Name, *CURLIB         Modul        giwssample       Name, generisch*, *PGM, *ALI         Bibliothek        zeiq       Name, generisch*, *PGM, *ALI         H für weitere Werte        Name, *LIBL, *CURLIB         Text       'Pessebreibung'       *ENTMODIXT	-
lext Beschreibung <u>*Enimobiki</u>	
Ende	
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F10=Zusätzl. Parameter	
F12=Abbrechen F13=Verwendung der Anzeige F24=Weitere Tasten	
	1/2
۲۰۲۶ International Internatio	

Programmobjekt erstellen

Für die Einbindung eines Webservices können durchaus auch mehrere Programme auf dem IBM i genutzt werden. Dazu ist es erforderlich, dass die einzelnen PCML-Anweisungen in ein einzelnes PCML-Dokument eingebunden werden. Dieses sollte das folgende Format aufweisen:

```
<program name="p1" ... > ... </program>
<program name="p2" ... > ... </program>
</pcml>
```



PCML-Dokumente können bei Bedarf auch manuell angepasst werden. Damit die Ergebnisse allerdings nicht für Verwunderung sorgen, müssen die Regeln für PCML genau eingehalten werden.

Wenn Sie unsicher im Umgang mit PCML sind, dann ist Ihnen die Verwendung der PCML-Dateien zu empfehlen, wie sie per Standard von den Kompilierbefehlen erzeugt werden.

Ab V6R1 werden die PCML-Information mit den Kompilierbefehlen erzeugt. Beispiele dazu sind:

```
CRTRPGMOD PGMINFO(*PCML *MODULE)
CRTCBLMOD PGMINFO(*PCML *MODULE)
```

Verwenden Sie die Betriebssystemversion V5R4 (Achtung: Spezielle PTFS sind hier erforderlich!), dann muss im Header-Bereich von RPG die folgende Codezeile vor dem Kompilieren eingegeben werden:

H PGMINFO(\*PCML:\*MODULE)

IBM liefert zusammen mit dem integrierten Webservice-Server Beispielanwendungen aus.

Für dieses sehr einfache RPG-Beispiel, das für Demozwecke in einem Webservice eingebunden werden kann und anhand dessen Sie sehen, wie ein Webservice mit einem RPG-Programm erstellt werden kann, wird ein Programm zur Umrechnung der Temperatur von Grad in Fahrenheit genutzt. Das dazugehörige Serviceprogramm QIWSSAMPLE befindet sich in der Bibliothek QSYS-DIR. Der Quellcode zu diesem Serviceprogramm befindet sich im Standardinstallationsverzeichnis für den Webservice-Server:

/QIBM/ProdData/OS/WebServices/V1/server/samples/ConvertTemp/ CNVRTTMP.RPGLE



Web-Programmierung

SOAP Webservices

5.5.7 Seite 8

**Kapite** 



RPG-Beispielprogrammcode im IFS

Schauen Sie sich den Beispielcode kurz an.

0	🖉 QIWSSAMPLE.RPGLE 🕱					
	Zeile 2	Spalte 1 Ersetzen				
		HKeywords++++++++++++++++++++++++++++++++++++				
E						
16	000101	h PGMINFO(*PCML:*MODULE) THREAD(*SERIALIZE)				
H	000152	d ConvertTemp pr				
	000154	d tempIn 10 const				
l Ŀ	000155	d tempOut 10				
H	000157	p ConvertTemp b export				
	000159	d ConvertTemp pi				
	000160	d tempIn 10 const				
H	000161	d tempOut 10				
	000163	d tempI s 8P 2				
	000164	d tempO s 8P 2				
	000165	d value S 50Å				
	000166	/free				
	000167	value = %STR(%ADDR(tempIn));				
	000168	<pre>temp1=%DEC(value:7:2);</pre>				
	000169	temp0 = (5/9)*(temp1 - 32);				
	000170	value = %CHAR(tempO);				
	000171 tempOut = value;					
	000172	<pre>%STR(%ADDR(tempOut):10)=tempOut;</pre>				
	000173	*inir = *on;				
	000174	/end-Iree				
	000175	p ConvertTemp e				

**RPG-Beispielcode** 

Das Beispiel ist wie die meisten neueren IBM-Codebeispiele in Free Format RPG geschrieben.



Web-Programmierung	tia i
SOAP Webservices	Ka
In der H-Bestimmung finden Sie die PCML-Informationen. Diese sind für die Verwendung des RPG-Programms als Webservice notwendig.	5.5.7 Seite 9
In den Prototypangaben finden Sie die beiden Parameter	

• tempIn und

• tempOut.

Die Rechenbestimmungen führen eine Berechnung aus, welche die Temperaturen Grad Celsius und Fahrenheit berechnen.

Da dieses Beispiel über die notwendigen PCML-Informationen verfügt, können Sie es nun als Webservice einbinden.

6

## 5.5.7 Seite 10

#### 5.5.7.2 SOAP Webservice erstellen

Die Erstellung des SOAP Webservices wird in der HTTP-Admin-Oberfläche vorgenommen. Hier finden Sie einen Assistenten, der Sie durch die Konfiguration führt. Wie dieser zu starten ist, ist zuvor beschrieben. Die weiteren Einzelschritte für die Erstellung des SOAP Webservices führen Sie nun aus.

Um dieses Serviceprogramm als Webservice nutzen zu können, geben Sie nun dessen Namen und die zugehörige Bibliothek in die Eingabemaske des Konfigurationsassistenten ein.

#### Achtung:

Der Konfigurationsassistent prüft, ob das angegebene Objekt tatsächlich in der Bibliothek existiert.

Bestätigen Sie die Eingabe mit einem Klick auf die Schaltfläche "Weiter".



RPG als Webservice angeben

Um ein solches ILE-Programm mit einem Webservice einsetzen zu können, müssen Sie dessen Namen und dessen Speicherort angeben. Basierend auf dem ILE- oder dem Serviceprogramm wird dann der Webservice erstellt. Für die Angabe eines Programmobjekts kann wahlweise dessen Name und die Bibliothek angegeben oder das IFS nach dem gewünschten Objekt durchsucht werden. Letztere Option kann jedoch viel Zeit in Anspruch nehmen und sollte deshalb nur im Notfall verwendet werden.

Mit einem Klick auf die Schaltfläche "Weiter" gelangen Sie in die nächste Anzeige des Konfigurationsassistenten.



## 5.5.7.2.1 Namensvergabe für den Webservice

Dem Webservice muss ein eindeutiger Name zugeordnet werden. Dieses wird in der nächsten Anzeige gemacht. Als Standardeinstellung verwendet der Assistent den Namen des Programms, welches als Webservice eingebunden werden soll. Demzufolge steht als Name im Beispiel QIWSSAMPLE in der Anzeige. Dieser Webservicename und auch die zugehörige Bezeichnung können natürlich überschrieben werden.

CHTTP-Server-Administration au	ASZEBIS - Windows Internet Explorer	
💽 🗢 🙋 http://aszebis:2001/HT	TPAdmin	🖌 🛃 🖾 Bing
<u>D</u> atei <u>B</u> earbeiten <u>A</u> nsicht <u>E</u> avoriten	Extras ?	
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene :	Sites 🔻 🙋 Web Slice-Katalog 👻	
HTTP-Server-Administration auf ASZEBI	5	🛅 🔹 🔝 🛸 🖃 🖶 👻 Sejte 🗸 Sigherheit 🗸 Extgras 🗸 🔞 🗸 🂙
IBM Web Administration for i Konfiguration Verwaltung Erw	eitert   Zugehörige Links	WebSphere.
Alle Server HTTP-Server   Appli	cation Server	
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li><sup>™</sup> Web Services Server erstellen</li> <li><sup>™</sup> HTT-Server erstellen</li> <li><sup>™</sup> Anwendungssener erstellen</li> <li><sup>™</sup> WebSphere Portal erstellen</li> </ul>	Web Services Server erstellen           Neuen Service implementieren:Namen für Service angeben - Schritt 5 von 9           Einen eindeutigen Namen für diesen Service angeben.           Servicename:         WSGRAD01           Servicebeschreibung:         WS Beispiel für Umrechnung Grad           Zurück         Weiter           Abbrechen	
		📢 Lokales Intranet 🦛 🔹 100% 👻 🚲

## Webservicename

Bestätigen Sie die Eingaben mit einem Klick auf die Schaltfläche "Weiter".



apitel

5.5.7 Seite 12

### 5.5.7.2.2 Auswahl der Exportprozeduren

Die Angabe der Parameter ist für Webservices in Verbindung mit dem aufrufenden Programm von besonderer Bedeutung. Der Webservice muss einen Einstiegspunkt für das anzubindende RPG-Programm haben. Für RPG-Programme wird in der nächsten Anzeige lediglich ein Eintrag angezeigt. Dabei handelt es sich um die Prozedur-Main, über die jedes RPG verfügt. Sollten Sie anstelle eines RPG-Programms ein Serviceprogramm einbinden, dann können durchaus auch mehrere Exportprozeduren angezeigt werden.

Mit dem Erweitern des Prozedureneintrags werden die dazugehörigen Parameter zur Auswahl angezeigt. Über eine Auswahl kann dieser als Eingabe-/Ausgabeparameter selektiert werden. Zudem wird die Art des Parameters dargestellt.

Anders als die RPG-Programme müssen Sie für den Webservice zwischen dem Eingabe- und dem Ausgabeparameter unterscheiden. Mögliche Angaben sind hier

- Eingabeparameter,
- Ausgabeparameter oder
- Eingabe-/Ausgabeparameter.

Damit die Parameter angezeigt bzw. verwaltet werden können, erweitern Sie die Anzeige mit einem Klick auf den gewünschten Serviceeintrag.

C HTTP-Server-Administration auf AS2	ZEBIS - Windows Internet Explorer	
COO - E http://aszebis:2001/HTTPAc	idmin	🖌 🗲 K 🔁 Bing
Datei Bearbeiten Ansicht Eavoriten Ex	<u>xtras ?</u>	
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene Sites	; 🔹 🙋 Web Slice-Katalog 👻	
HTTP-Server-Administration auf ASZEBIS		🏠 🔻 🖾 🕤 🚍 🖛 Y Seite Y Sigherheit Y Extyras Y 🔞 Y 🎽
IBM Web Administration for i	rt   Zugehörige Links	WebSphere.
Alle Server HTTP-Server   Applicatio	on Server	
Allgemeine Tasks und Assistenten     Web Services Sener erstellen     Anwendungssener erstellen     WebSphere Portal erstellen     WebSphere Zottal erstellen	portierte Prozeduren sind Eingangspunkte in ein Programmobjekt, die Wet genstandiger Anweisungen in einer höheren Programmiersprache, die eine rückkeht. Ein Sewiceprogramm enthält mindestens eine Prozedur. Ein Pro- e nachfolgende Tabelle enthält alle exportierten Prozeduren, die in dem Pro- rückkeht. Ein Sewiceprogramm enthält mindestens eine Frozedur. Ein Pro- e nachfolgende Tabelle enthält alle exportierten Prozeduren, die in dem Pro- rückkeht. Ein Sewiceprogramm enthält mindestens eine Frozeduren, die in dem Pro- e nachfolgende Tabelle enthält alle exportierten Prozeduren, die in dem Pro- stauk weiche Daten von Cliente gesendet werden und was vom Viels-Sewices führ reportprozeduren: wordprozeduren vom Verwendung v v QWVSSAMPLE v CONVERTTEMP B TEMPIN B TEMPOUT input/vortput v dies auswählen (Alles abwählen) Alles einbliende input/output durput Weiter (Abbrechen) report	Service-Operationen zugeordnet werden. Eine Prozedur besteht aus einer Gruppe bestemmte Task ausführt und anschließend zum aufrufenden Programm igramm enthält nur eine einzige Prozedur. bogrammebjekt enthälten sind. das durch diesen Web-Senice ausgelagert werden e Prozedurparameter zu andern. Das Parameterathiout Verwendung hat Einfluss e zurückgegeben wird. Für Feldgruppentypparameter kann eine Anderung des en. batentyp Anzahl en corpx forpet
Fertig		Sector States Intranet 🛛 🖓 🔹 🔍 100% 👻 🧾





Die in diesem Bereich zu tätigenden Selektionen sind folgende:

- Mit dem Auswahlfeld "Auswählen" legen Sie fest, welche Prozeduren für den Webservice genutzt werden können. Hier können zwar durchaus auch solche Prozeduren selektiert werden, die im Webservice eigentlich nicht zum Einsatz kommen, doch ist dies nicht zu empfehlen.
- Mit der Auswahl des Parameters legen Sie den Parameter fest, der für das Programm in Zusammenarbeit mit dem Webservice genutzt werden soll.
- In der Spalte "Verwendung" geben Sie an, ob der betreffende Parameter als Eingabe- und/oder als Ausgabeparameter zum Einsatz kommt. Bei der Auswahl des Parameters ist unbedingt darauf zu achten, dass der Webservice diesen entsprechend bedient.

Neben den Auswahlfeldern gibt es noch zwei weitere Informationsfelder in dieser Anzeige. Diese geben Auskunft über den Datentyp des jeweiligen Parameters.

Der Eintrag "Anzahl" liefert Informationen darüber, ob es sich beim Parameter um eine Feldgruppe handelt. Ist dies der Fall, dann ist in diesem Bereich ein Wert enthalten, der die Anzahl der Einträge in der Feldgruppe liefert.

Hier ist der Parameter TEMPIN als Eingabeparameter definiert und der Parameter TEMPOUT als Ausgabeparameter. Dies hat zur Folge, dass der Webservice-Client lediglich den TEMPIN-Parameter übergeben muss, der als Ergebnis die Rückgabe des Parameters TEMPOUT erhält.

Exportprozeduren: 😮 Auswählen Prozedurname/Parametername Verwendung Datentyp Anzahl QIWSSAMPLE ~ CONVERTTEMP TEMPIN char input TEMPOUT char output

Folglich sollten die Parametereinstellungen so aussehen:

#### Angepasste Parameter

Ergänzung 44/2018

Die Konfiguration wird mit einem Klick auf die Schaltfläche "Weiter" fortgesetzt.

Alles einblenden Alles ausblenden

Augobaseta Daugunatan

Alles auswählen Alles abwählen





Seite 13

## 5.5.7 Seite 14

#### 5.5.7.2.3 Angabe der Benutzer-ID

Jeder Job muss mit einem vorgegebenen Benutzer auf dem IBM i ausgeführt werden, wozu in der Folgeanzeige eine gültige Benutzer-ID anzugeben ist. Wahlweise kann es sich hierbei in der nächsten Anzeige um den Standardbenutzer für die Webservices oder eine individuelle Benutzer-ID handeln. Natürlich gelten dabei auch die allgemeinen Regeln für eine Benutzer-ID. Das heißt, sie muss aktiv sein und über ausreichende Berechtigungen für die gewünschten Tätigkeiten verfügen.

C HTTP-Server-Administration au	f ASZEBIS - Windows Internet Explorer		
🔆 🕞 🗢 🔊 http://aszebis:2001/H	TTPAdmin	V 🛃 🗙 🔽 Bing	<b>P</b> •
Datei Bearbeiten Ansicht Eavoriten	Extras ?		
🚖 Favoriten 🛛 🚔 🏉 Vorgeschlagene	Sites 🔹 🙋 Web Slice-Katalog 🔹		
HTTP-Server-Administration auf ASZEB	IS	🏠 👻 🖾 🛸 🖃 🖶 👻 Sejte 🗸 Sigherheit 👻 Extra:	• @• »
IBM Web Administration for i Konfiguration Verwaltung Erv	veitert   Zugehörige Links	WebSphere,	IBM
Alle Server HTTP-Server   Appl	ication Server		
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li>♥ Web Services Server erstellen</li> <li>♥ HTTP-Server erstellen</li> <li>♥ Anwendungsserver erstellen</li> <li>♥ WebSphere Portal erstellen</li> </ul>	Web Services Server erstellen Neuen Service implementieren:Benutzer-ID für diesen Service anger Zur Ausführung des Programmobjekts, das die Web-Service-Logik enth	ben - Schritt 7 von 9 ält, benötigt der Service eine OS-Benutzer-ID.	
	Benutzer-ID für diesen Service angeben: 2 O Die Benutzer-ID des Servers verwenden		
	Die Benutzer-ID des Servers benötigt die erforderliche Berechtigu	ng für dieses und alle zusätzlichen Programmobjekte.	
	C Eine vorhandene Benutzer-ID angeben		
	Zurück Weiter Abbrechen		
			V
		Schales Intranet 🖓 🔹 🍕	100% 🔹 🛒

Servicebenutzer-ID

Für das Beispiel wird die Benutzer-ID des Servers verwendet und auf die Schaltfläche "Weiter" geklickt.



Seite 15

Web-Programmierung

TRM

O

SOAP Webservices

## 5.5.7.2.4 Angabe der Bibliotheksliste

CHTTP-Server-Administration auf ASZEBIS - Windows Internet Explorer

Allgemeine Tasks und Assistenten Web Services Server erstellen

HTTP-Server-Administration auf ASZEBIS

Datei Bearbeiten Ansicht Eavoriten Extras ? 🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene Sites 🔹 🙋 Web Slice-Katalog 🔹

Alle Server HTTP-Server | Application Server

IBM Web Administration for i Konfiguration Verwaltung Enweitert | Zugehörige Links

Analog zu klassischen Anwendungen muss für den Aufruf der RPG-Programme eine entsprechende Umgebung definiert werden. Die Wahl der Bibliotheksliste ist hier ebenso notwendig wie im nativen Ausführungsbereich.

In der nächsten Anzeige legen Sie die Bibliotheken fest, die für die Ausführung des Webservices benötigt werden. Durch die Selektionsoptionen kann zudem festgelegt werden, an welche Stelle des Benutzerteils der Bibliotheksliste der einzugebende Name der Bibliothek gesetzt werden soll.

## Specify library list position for this Web service: O Insert libraries in front of user library portion of the library list Insert libraries at the end of user library portion of the library list Bibliothekslisteneinträge: 📀 Bibliotheksname O ZEIG Hinzufügen Alles entfernen Zurück Weiter Abbrechen

IBM POWER Systems i Softwareentwicklung und -modernisierung

## Bibliotheksliste anpassen

Nachdem die gewünschten Bibliothekseinträge hinzugefügt worden sind, setzen Sie die Arbeit mit einem Klick auf die Schaltfläche "Weiter" fort.

Damit sind die einzelnen Schritte in der Definition des Webservices abgeschlossen. Dieser ist allerdings noch nicht abschließend erstellt. In der Folgeanzeige sehen Sie eine Zusammenfassung der bisher getätigten Angaben.



💌 🐓 🗙 🔁 Bir

🏠 🔹 🔝 🕤 🖶 🔹 Seite 🔹 Sigherheit 🔹 Extras 🔹 🔞 🔹

(?) WebSphere.



## 5.5.7 Seite 16

#### 5.5.7.2.5 Zusammenfassung

Die Zusammenfassung der Konfigurationsangaben beinhaltet alle Teilschritte, die Sie zuvor definiert haben. Prüfen Sie diese sorgfältig! Natürlich können einzelne Angaben auch später noch angepasst werden.

In der Serverübersicht erscheint der neue Webservice-Server. Zusätzlich finden Sie die Informationen zum automatisch miterstellten Webserver. Neben dessen Namen ist auch der Port wichtig, über den der Server erreicht werden kann.

C HTTP-Server-Administration au	f ASZEBIS - Windows Intern	et Explorer	
🔆 🕞 🔻 🙋 http://aszebis:2001/HT	TPAdmin	v 49	× Bing
Datei Bearbeiten Ansicht Eavoriten	Extras ?		
🚖 Favoriten 🛛 👍 🏉 Vorgeschlagene :	Sites 🔹 🙋 Web Slice-Katalog 🔹		
HTTP-Server-Administration auf ASZEB1	15	🛅 •	🔝 🗉 🖶 🔹 Seite 🔹 Sigherheit 🔹 Extras 🛛 🔞 👻
IBM Web Administration for i Konfiguration Verwaltung Erw Alle Server HTTP-Server   Appli	eitert   Zugehörige Links		() WebSphere.
Allgemeine Tasks und Assistenten     Web Services Server erstellen     HTTR-Sperver erstellen     Anvendungsserver erstellen     WebSphere Portal erstellen	Web Services Serve Zusammenfassung - Schrit Wenn Sie auf Fertig stellen i Server Services Operat Informationen über Web Se Server-Beschreibung: Interner Port-Bereich: Server-Root Server-Root Server-Rot Server-Rot Server-Rot Server-Rot Server-Rot Server-Rot Server-Rot Server-Rot Server-Beschreibung: Interner Port-Bereich Server-Rot Server-Rot Server-Rot Server-Rot Server-Rot Server-Rot Server-Beschreibung:	r erstellen f 9 von 9 flicken, wird alles erstellt, was zur Ausführung des Web Services Server erfor ionen mices Server DEMOWS01 Web Service Server für Test 10044 - 10053 www.DEMOWS01 http://assebis.epis.de/10054 OWSERVICE Web	derlich ist Druckbare Zusammenfassung
Fertig			Lokales Intranet

Zusammenfassung Allgemein

Im Bereich "Services" finden Sie die zuvor definierten Services. Diese werden mit den folgenden Informationen dargestellt:

- Name,
- Beschreibung,
- Installationspfad,
- Benutzer-ID, unter der der Service ausgeführt wird,
- eingebundenes RPG-Programm,
- Bibliotheksliste, die für den Service genutzt wird, und
- URL für den Webservice.





Seite 17



Zusammenfassung Services

Im Tabellenbereich "Operationen" finden Sie die Webservice-Operationen, die für diese Prozedur zum Einsatz kommen.

C HTTP-Server-Administration au	f ASZEBIS - Windows Internet Explorer			
💽 🗢 🔊 http://aszebis:2001/H1	TTPAdmin		🖌 🗲 🗙 🔁 Bing	P •
Datei Bearbeiten Ansicht Eavoriten	E <u>x</u> tras <u>?</u>			
🚖 Favoriten 🛛 👍 🏉 Vorgeschlagene :	Sites 🔹 🙋 Web Slice-Katalog 🔹			
HTTP-Server-Administration auf ASZEB	15		🚹 • 🖾 - 🖬 •	Seite - Sigherheit - Extras - 🔞 - 🂙
IBM Web Administration for i Konfiguration Verwaltung Erw Alle Server HTTP-Server   Appli	eltert   Zugehörige Links ication Server			WebSphere.
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li>♥ Web Services Server erstellen</li> <li>♥ HTTB-Server erstellen</li> <li>♥ Anwendungsserver erstellen</li> <li>♥ WebSphere Portal erstellen</li> <li>♥ WebSphere Portal erstellen</li> </ul>	Web Services Server erstellen         Zusammenfassung - Schrift 9 von 9         Wenn Sie auf Fertig stellen klicken, wird alles ers         Server       Services         Operationen für "QIVSSAMPLE,PGI         Operationsnamen         qiwssample         qiwssample_XIIL	tellt, was zur Ausführung des Web Senvices S M* :	erver erforderlich ist.	Druckbare Zusammenfassung
Fertig			Scheles Intranet	va • 🔍 100% • 💡

### Zusammenfassung Operationen

Sind die Angaben korrekt, dann erstellen Sie den Webservice mit einem Klick auf die Schaltfläche "Fertig stellen". Nun werden die einzelnen Konfigurationsschritte ausgeführt.



Kapitel

## Web-Programmierung

SOAP Webservices

## 5.5.7 Seite 18



### Fertigstellung

Das Ergebnis sollte ein einsatzbereiter Webservice sein.

Konnte der Service fehlerfrei angelegt werden, werden der Webservice-Server und der zugehörige Webserver automatisch gestartet. Den Status sehen Sie in der nächsten Anzeige. Natürlich werden die beiden Server auch in der allgemeinen Serverübersicht mit aufgeführt und können von dort verwaltet werden.









Je nach System kann der Serverstart einen Augenblick dauern.

Damit sind die für die Anlage eines Webservices notwendigen Teilschritte abgeschlossen, und der Webservice kann genutzt werden.

Im linken Navigationsbereich der Anzeige finden Sie unter anderem den Eintrag "Implementierte Services verwalten".



Implementierte Services verwalten

Mit dieser Auswahl gelangen Sie in den Übersichtsbereich der definierten Webservices, der als zentrale Administrationseinheit für die Webservices genutzt wird.



Kapitel

## Web-Programmierung

## 5.5.7 Seite 20

C HTTP-Server-Administration auf	f ASZEBIS - Windows Inte	ernet Explorer				
🔆 🔊 🗢 🙋 http://aszebis:2001/HT	TTPAdmin			🖌 🗲 🗙 🔁 Bing	P -	
Datei Bearbeiten Ansicht Eavoriten	E <u>x</u> tras <u>?</u>					
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene S	Sites 🝷 🙋 Web Slice-Katalog	-				
HTTP-Server-Administration auf ASZEBI	15				🏠 🔹 🗟 🐇 🖃 🚔 🔹 Seijo	e 🔹 Sigherheit 👻 Extras 🕶 🕡 👻 🎽
IBM Web Administration for i Konfiguration Verwaltung Erwe	eitert   Zugehörige Links					(?) WebSphere. IBM
Alle Server   HTTP-Server Applic	ation Server					
🛚 Gestartet 📄 🔲 💋 Server: 🛛	DEMOWS01 - V1.3 (Web	-Services) 🔽				
▼ Allgemeine Tasks und Assistenten <sup>1</sup> Web Services Server erstellen <sup>1</sup> HTTP-Server erstellen	DEMOWS01 > Implement	ierte Services verv <b>rvices verwa</b>	valten Iten			
U WebSphere Portal erstellen	Stand vom 13.08.2011 07	43:06.				
<ul> <li>Web-Service-Assistenten</li> <li>Neuen Service implementieren</li> </ul>	Implementierte Services:	0				
<ul> <li>Server-Eigenschaften</li> </ul>	Servicename	Status	Starttyp	WSDL - Servicedefinition		
D Eigenschaften D Sonor Tracing	ConvertTemp	Gestartet A	Automatisch	Definition anzeigen		
B HTTP-Server anzeigen	O WSGRADUT	Gestaltet A	Automatisen	1 Definition anzeigen		
<ul> <li>✓ Services</li> <li>▲ Implementierte Services verwalten</li> </ul>	Implementieren Ak	tualisieren				
✓ Web Performance © Web Performance Advisor						
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> <li>Web Log Monitor</li> <li>Protokolle anzeigen</li> </ul>	Schließen					
<						×
					Schales Intranet	🖓 • 🔍 100% • 💡

Die verfügbaren Services



# **E**

Seite 21

## 5.5.7.3 Verwaltung der definierten Webservices

In der Anzeige "Implementierte Services verwalten" finden Sie die Übersicht über die definierten Webservices. Diese können von der zentralen Stelle aus verwaltet werden.

Damit dies für einen einzelnen Webservice erfolgen kann, muss er in der Übersicht markiert werden. Nach dem Markieren ändert sich die Anzeige, und es erscheinen unter anderem die Auswahlmöglichkeiten zum Starten bzw. Stoppen des Webservices.

C HTTP-Server-Administration auf	ASZEBIS - Windows Internet Explorer	
💽 🗢 🙋 http://aszebis:2001/HT	TPAdmin	🖌 🔄 🖌 🔁 Bing
Datei Bearbeiten Ansicht Eavoriten	Extras ?	
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene S	Sites 👻 🖉 Web Slice-Katalog 👻	
HTTP-Server-Administration auf ASZEBI	5	🏠 🔻 🖾 🗉 🖶 🥆 Seite - Sigherheit - Extras - 🕢 - 🎽
IBM Web Administration for i Konfiguration Verwaltung Erwe	eitert   Zugehörige Links	((?) WebSphere.
Alle Server   HTTP-Server Applic	ation Server	
🗣 Gestartet 🕞 🔲 🗭 Server:	DEMOWS01 - V1.3 (Web-Services) 💌	
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li>♥ Web Services Server erstellen</li> <li>♥ HTTP-Server erstellen</li> <li>♥ Anwendungsserver erstellen</li> <li>♥ WebSphere Portal erstellen</li> </ul>	DEMOWS01 > Implementierte Services verwalten Implementierte Services verwalten Stand vom 13.08.2011.07.44.47	
▼ Web-Service-Assistenten <sup>3</sup> Neuen Service implementieren	Implementierte Services: 0	
▼ Server-Eigenschaften	Servicename         Status         Starttyp         WSDL - Servicedefinition           O         ConvertTemp         Gestartet         Automatisch         Definition anzeigen           O         WSGRAD01         Gestartet         Automatisch         Definition anzeigen	
▼ Services	Implementieren Stoppen Eigenschaften Deinstallieren Aktualisieren	Service testen
✓ Web Performance ♥ Web Performance Advisor		
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> <li>Web Log Monitor</li> <li>Pubersicht anzeigen</li> </ul>	Schließen	
		×
		Suckales Intranet 🦓 🔹 🔍 100% 👻 🔐

Einzelnen Webservice verwalten

Die Webservices lassen sich von dieser Stelle aus starten, stoppen, testen und bei Bedarf auch deinstallieren (vgl. folgende Abbildung).



Ergänzung 44/2018



Web-Programmierung

SOAP Webservices

5.5.7 Seite 22 Unter anderem erhalten Sie mit der Option "Definition anzeigen" einen Einblick ins Innenleben des Webservices. Mit der Erstellung des Webservices hat der Assistent die eingegebenen Informationen in XML umgesetzt. Das Beispielergebnis sieht folgendermaßen aus:

Chttp://aszebis:10054/web/services/WSGRAD01?wsdl -	Windows Internet Explorer	
COO V Intp://aszebis:10054/web/services/WSGRAD01?ws	dl 💌 🔂 🕅 🖳 Bing	P •
Datei Bearbeiten Ansicht Eavoriten Extras ?		
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene Sites 🔹 🖉 Web Slice-Katal	og •	
<pre>// http://aszebis:10054/web/services/WSGRAD01?wsdl</pre>	🏠 🔻 🔝 🐇 🖃 👼 👻 Sejte – Sigherheit -	Extras 🔹 🔞 🔹 🎽
		~
xml version="1.0" encoding="UTF-8" ?		
<ul> <li><wsdi:definitions http:="" li="" org.apacho.axis2="" xm<="" xmins:wsdi="http://schemas.x&lt;br&gt;xmins:ns1=" ysd"=""> </wsdi:definitions></li></ul>	tmisoap.org/wsdi/" xmins:axis2="http://wsgrad01.wsbeans.iseries"	
xmins:http="http://schemas.xmisoap.org/w	rsdl/http/" xmlns:ns0="http://wsgrad01.wsbeans.iseries/xsd"	
xmlns:xs="http://www.w3.org/2001/XMLSc	hema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"	
xmlns:mime="http://schemas.xmlsoap.org/v	wsdl/mime/" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"	
targetNamespace="http://wsgrad01.wsbean	s.iseries">	
<pre><wsdl:documentation>WSGRAD01</wsdl:documentation></pre>	nentation>	_
- <wsdi:types></wsdi:types>	hears isories (yed" attributeFormDefault="gualified" elementFormDefault="gualified"	
- cxs:scriena xmins:ns= http://wsgrad01.ws targetNamespace="http://wsgrad01.wsb	peans.iseries/xsd attributeromberauit= quaimed elementromberauit= quaimed	
- <xs:element name="convertemp"></xs:element>		
- <xs:complextype></xs:complextype>		
- <xs:sequence></xs:sequence>		
<xs:element minoccurs="0" name="pa&lt;/th&gt;&lt;th&gt;aram0" nillable="true" type="ns:CONVERTTEMPInput"></xs:element>		
- <xs:complextype <="" name="CONVERTTEMPIn" p=""></xs:complextype>	put">	
- <xs:sequence></xs:sequence>		
<pre><s:element minoccurs="0" name="_IE&lt;/pre&gt;&lt;/th&gt;&lt;th&gt;MPIN nillable=" true"="" type="xs:string"></s:element></pre>		
<ul> <li></li> <li></li></ul> <li></li> <li><th>e"&gt;</th><td></td></li>	e">	
- <xs:complextype></xs:complextype>		
- <xs:sequence></xs:sequence>		
<xs:element minoccurs="0" name="re&lt;/th&gt;&lt;th&gt;.turn" nillable="true" type="ns:CONVERTTEMPResult"></xs:element>		
- <xs:complextype <="" name="CONVERTTEMPRe" p=""></xs:complextype>	isult">	
- <xs:sequence></xs:sequence>		
<pre><xs:element minoccurs="0" name="_TE&lt;/pre&gt;&lt;/th&gt;&lt;th&gt;MPOUT" nillable="true" type="xs:string"></xs:element></pre>	<b>~</b>	
Fertig	Second Se	• 🔍 100% •

XML des Webservices

Damit sind die Konfigurationsschritte für das Bereitstellen eines RPG-Programms als Webservice abgeschlossen.



Web-Programmierung SOAP Webservices

Kapitel

5.5.7

Seite 23

IBM stellt zudem einen Webservice-Testclient bereit, der automatisch mit der Auswahl "Service testen" gestartet wird.



Service testen

Der Testclient startet in einem separaten Browserfenster.

🖉 Web Services Explorer - Windows Internet Explore		
🕒 🗢 🖉 http://aszebis:2006/wsexplorer/wsexplorer.jsp	2wsdl=http://aszebis:10054/web/services/W5GRAD01?wsdl	P •
Datei Bearbeiten Ansicht Eavoriten Extras ?		
🚖 Favoriten 🛛 🚖 🏉 Vorgeschlagene Sites 🔹 🛃 Web Slice-K	atalog 🕶	
C Web Services Explorer	🟠 🔻 🔂 🕆 🖃 🌧 👻 Seite 🛩 Sigherheit 👻 Extr	'as 🔻 🔞 🕶 🦇
i5/0S Web Services Test Client		
९५- Navigator 🔗 🖉	Actions	Q
Butto://assebs:10054/web/services/WSGRAD012/wsdl         C: @ http://assebs:10054/web/services/WSGRAD012/wsdl         D: @ WSGRAD013         D: @ http://assebs:10054/web/services/WSGRAD012/wsdl         D: @ WSGRAD0130AP1Binding         D: @ WSGRAD0120AP1Binding         D: @ WSGRAD0120AP1Binding         D: @ Convertiemp_3NL         D: @ qivesample	Status       Invoke a WSDL Operation     Sol       Enter the parameters of this WSDL operation and click Go to invoke.       Endpoints       Inter/desceles: 10054/web/services/WSGRADO1 v       * somestients       * status       Inter is nothing to be displayed in the form view. Please switch to the source view for the SOAP request and response.	
	Subales tritranet	100% +

Webservice-Testclient



## Web-Programmierung

SOAP Webservices

## 5.5.7 Seite 24

Wenn Sie den Eintrag "converttemp" in der linken Navigationsleiste auswählen, werden im Hauptfensterbereich die Parameter für den Webservice angezeigt. Um die Parameter und deren Inhalte angeben zu können, müssen Sie auf "add" klicken.

Mit einem Klick auf "Go" wird der Webservice ausgeführt.

Alternativ zum integrierten Testclient können Sie auch andere Testclients für den Webservice nutzen. Als Beispiel ist SOAPUI zu nennen.

Die Webservice-Serverinstanz ist nun erstellt. Zudem wurde diesem Server ein Webservice hinzugefügt. Natürlich kann ein einzelner Server auch mehrere Webservices zeitgleich bedienen, und um diese hinzufügen zu können, klicken Sie in der Verwaltungsübersicht im Navigationsbereich auf den Eintrag "Neuen Service implementieren".

🖉 HTTP-Server-Administration auf ASZEBIS - Windows Internet Explorer 📃 🔲 🔀					
🚱 🗢 🙋 http://aszebis:2001/H1	TTPAdmin		✓ 47 ×	Bing P -	
Datei Bearbeiten Ansicht Eavoriten	E <u>x</u> tras <u>?</u>				
🖕 Favoriten 🛛 🖕 🏉 Vorgeschlagene S	Sites 🔹 🙋 Web Slice-Katalog 🝷				
HTTP-Server-Administration auf ASZEBI	15		🟠 • 🔊	- 📑 🖶 - Seite - Sigherheit - Extras - 🕡 - 🎽	
IBM Web Administration for i Konfiguration Verwaltung Erw	eitert   Zugehörige Links			() WebSphere.	
Alle Server   HTTP-Server Applic	ation Server				
🛚 Gestartet 下 🗖 🛃 Server:	DEMOWS01 - V1.3 (Web-Services) 💌				
<ul> <li>✓ Allgemeine Tasks und Assistenten</li> <li><sup>1</sup> Web Services Server erstellen</li> <li><sup>1</sup> HTTP-Server erstellen</li> <li><sup>1</sup> Anwendungsserver erstellen</li> <li><sup>2</sup> WebSphere Portal erstellen</li> </ul>	DEMOWS01 > Implementierte Services Implementierte Services ver Standuom 12.08.2011.07.55.40	erwalten v <b>alten</b>			
▼ Web-Service-Assistenten <sup>1</sup> 2 Neuen Service implementieren	Implementierte Services; 2				
▼ Server-Eigenschaften	Servicename Status	Starttyp WSDL -	Servicedefinition		
B Eigenschaften B Server-Tracing	O WS001	Automatisch M Defini	ion anzeigen		
B HTTP-Server anzeigen	Wiscon Wiscon Wiscon States	Illert Automatisch 🖄 Defini	ion anzeigen		
▼ Services	Implementieren Stoppen Ei	enschaften Deinstallieren	Aktualisieren Service teste	n	
implementierte Services verwalten				-	
<ul> <li>Web Performance</li> <li>Web Performance Advisor</li> </ul>					
<ul> <li>Fehlerbestimmung</li> <li>Protokolle anzeigen</li> <li>Web Log Monitor</li> <li>Übersicht anzeigen</li> </ul>	Schließen				
< I III III III III III III III III III				1	
Fertig			Sector Se	es Intranet 🋛 🖓 👻 🔩 100% 👻 🛒	

Neuen Service implementieren

Damit wird der Konfigurationsassistent für die Bereitstellung des Webservices gestartet. Die notwendigen Schritte hierzu kennen Sie bereits.

